Utilisation des processus markoviens en reconnaissance de l'écriture

Use of Markov Processes in Writing Recognition

par Abdel BELAÏD et George SAON

Bât Loria, Campus scientifique, B.P. 239
54506 Vandœuvre-Lès-Nancy Cedex, France
Email : abelaid,saon@loria.fr

résumé et mots clés

Dans cet article, nous présentons une étude sur l'emploi de différents types de modèles de Markov en reconnaissance de l'écriture. La reconnaissance est obtenue par calcul de la probabilité *a posteriori* de la classe d'une forme. Ce calcul fait intervenir plusieurs termes qui, suivant certaines hypothèses de dépendance liées à l'application traitée, peuvent se décomposer en probabilités conditionnelles élémentaires. Si l'on suppose que la forme suit un processus stochastique uni- ou bidimensionnel qui de plus vérifie les propriétés de Markov, alors la maximisation locale de ces probabilités permet l'atteinte d'un maximum de la vraisemblance de la forme. Nous avons étudié plusieurs cas de conditionnement des probabilités élémentaires des sous-formes. Chaque étude est accompagnée d'illustrations pratiques relatives au domaine de la reconnaissance de l'écriture imprimée et/ou manuscrite.

Modèles de Markov, Champs aléatoires, Processus stochastiques, Reconnaissance automatique de l'écriture.

abstract and key words

In this paper, we present a brief survey on the use of different types of Markov models in writing recognition. Recognition is done by a posteriori pattern class probability calculus. This computation implies several terms which, according to the dependency hypotheses akin to the considered application, can be decomposed in elementary conditional probabilities. Under the assumption that the pattern may be modeled as a uni- or two-dimensional stochastic process (random field) presenting Markovian properties, local maximisations of these probabilities result in maximum pattern likelihood. We have studied throughout the article several cases of subpattern probability conditioning. Each case is accompanied by practical illustrations related to the field of writing recognition.

Markov Models, Random fields, Stochastic Processes, Writing Recognition.

1. introduction

Les modèles de Markov connaissent actuellement un essor important en reconnaissance des formes grâce à leur capacité d'intégration du contexte et d'absorption du bruit [Rab 89, Bel 94d]. Dans ces modèles, les formes sont décrites par une séquence de primitives qui seront observées dans les états du modèle. La probabilité d'émission de la forme par le modèle est calculée en maximisant, sur l'ensemble des chemins d'états, la probabilité d'observation des segments pondérée par les probabilités de transitions entre états. Ce calcul se fait généralement par maximum de vraisemblance. Le calcul de la vraisemblance de la forme par rapport au modèle intervient dans la règle de Bayes qui inclue la probabilité *a priori* du modèle.

Le but de cet article est d'étudier l'apport des modèles de Markov en reconnaissance automatique de l'écriture (RAE). Tout comme la parole, l'écriture se prête à une modélisation stochastique, à tous les niveaux de reconnaissance : morphologique, lexical et syntaxique [Sao 94, Bel 94b, Sao 95]. En effet, les lettres sont régies par un contexte lexical des mots de la langue se traduisant par

des probabilités d'apparition et de succession dans des mots. Ces probabilités peuvent être estimées à partir de statistiques concernant les fréquences d'occurrences de certains groupes de lettres. Cette modélisation stochastique trouve beaucoup d'intérêt dans la reconnaissance de l'écriture cursive où l'estimation probabiliste se fait sur des hypothèses de segmentation du mot en lettres ou en parties de lettres (graphèmes).

L'accent sera mis sur les différentes manières d'interpréter les termes liés par la formule de Bayes. En effet, la décomposition de ces termes peut conduire à des modélisations markoviennes différentes suivant les hypothèses d'indépendance permises par les applications.

Cet article est décomposé en deux parties. La première partie est centrée sur la modélisation de l'écriture par des modèles de Markov unidimensionnels à partir des différentes interprétations du calcul de la vraisemblance. Des exemples de systèmes existants viennent appuyer ces interprétations dans chacun des cas traités. Ces modèles restent malgré tout limités car ils transforment l'image des mots sous forme d'une séquence de symboles. Le défi de la recherche actuelle dans ce domaine consiste à prendre en compte les caractéristiques bidimensionnelles de l'écriture, c'est-à-dire à étendre les modèles de Markov au domaine plan. Nous montrerons dans la deuxième partie de cet article quelques avancées sur les modèles bidimensionnels de type PHMM (pseudo- ou planar-HMM) et sur les champs de Markov. Nous verrons que les principes de base relatifs à l'interprétation du calcul de vraisemblance restent maintenus.

Cadre de l'étude : Nous limiterons notre étude essentiellement aux processus markoviens discrets. Cependant, nous donnerons un exemple d'application des HMMs semi-continus à la RAE dans le paragraphe 2.3.2.

La plupart des modèles analysés, que ce soient des modèles uni- ou bidimensionnels, font appel à un apprentissage des paramètres de type maximum de vraisemblance (MLE pour Maximum Likelihood Estimation) par ré-estimation Baum-Welch, dérivée d'une estimation bayésienne dans laquelle le biais sur les modèles est supposé uniforme. Il existe néanmoins une classe non-négligeable de modèles markoviens qui se basent sur un apprentissage de type maximum d'information mutuelle (MMI) [Bah 86, Mer 88, Gop 89] ou maximum d'information discriminante (MDI) [Eph 87]. La première catégorie d'apprentissage est intrinsèquement liée à l'estimation maximum a posteriori (MAP) de la probabilité des modèles par rapport à la forme en considérant que les termes qui interviennent dans la formule de Bayes sont étroitement corrélés. L'inconvénient majeur de ce type d'approche reste le fait qu'il n'existe pas de formules de ré-estimation itérative comme dans le cas MLE. Ces techniques font généralement appel à des méthodes de type *descente du gradient* [Nor 94] qui tiennent peu compte des spécificités des modèles markoviens. Ces raisons ajoutées à l'absence de systèmes de RAE basés sur ces approches nous ont amenés à les écarter de notre étude.

De même, les problèmes d'apprentissage à partir de données incomplètes et leur solutions, comme la liaison de paramètres (*parameter tying*) [Bah 83], l'interpolation avec effacement (*deleted interpolation*) [Jel 80], ou encore l'apprentissage correctif (*corrective training*) [Bah 88] qui tiennent plutôt du domaine de l'implémentation pratique, ne seront pas traités dans cet article.

2. modèles de Markov 1D

2.1. définitions préliminaires

Définition 1

Soit K un ensemble d'indices. Un *processus stochastique* est une famille $(X_k, k \in K)$ de variables aléatoires définies sur un espace Ω. Il est dit discret si les variables aléatoires sont en nombre fini ou dénombrable : $K = \{1, \ldots, T\}$

Définition 2

Une variable aléatoire est dite discrète si elle prend ses valeurs dans un ensemble fini ou dénombrable. On notera par $P(X) \triangleq P(X = x)$ la probabilité (au sens générique) d'une réalisation x de la variable aléatoire X.

Définition 3

Une *chaîne de Markov* discrète d'ordre n est un processus stochastique discret avec des variables aléatoires discrètes (dont les réalisations sont appelées *états*), vérifiant la propriété de Markov :

$$P(X_t = s_{i_t} | X_{t-1} = s_{i_{t-1}}, \ldots, X_1 = s_{i_1}) = \\ = P(X_t = s_{i_t} | X_{t-1} = s_{i_{t-1}}, \ldots, X_{t-n} = s_{i_{t-n}}), \quad (1) \\ \forall t \in \{1, \ldots, T\}, \qquad s_{i_1}, \ldots, s_{i_T} \in S$$

où $S = \{s_1, \ldots, s_N\}$ représente l'ensemble des états.

Définition 4

Une chaîne de Markov d'ordre 1 est *stationnaire* si pour tout t et k :

$$P(X_t = s_i | X_{t-1} = s_j) = P(X_{t+k} = s_i | X_{t+k-1} = s_j) \quad (2)$$

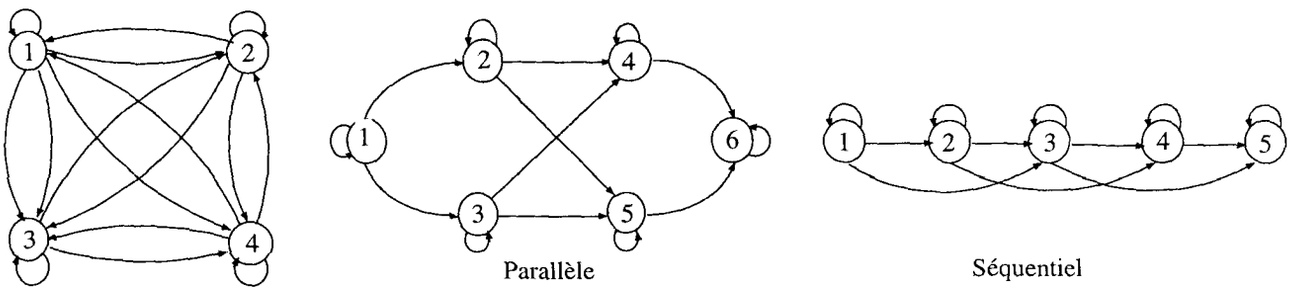
2.2. modèles de Markov cachés

Par la suite, on nommera *observation* toute trace quantifiable à un instant donné d'un processus physique quelconque.

Définition 5

Un *modèle de Markov caché* (Hidden Markov Model ou HMM) est une chaîne de Markov stationnaire où l'observation est une fonction probabiliste de l'état.

Le modèle résultant est un processus doublement stochastique où la composante relative à la suite d'états est cachée. On



(a) Ergodique

(b) Gauche-droite

Figure 1. – Exemples d'architectures de HMM.

désignera par $O = o_1 o_2 \dots o_T$ la suite d'observations et par $Q = q_1 q_2 \dots q_T$ la suite (inconnue) des états.

Définition 6

Un HMM discret du premier ordre est défini par :

$S = \{s_1, s_2, \dots, s_N\}$, l'ensemble des N états du modèle. On désigne un état au temps t par $q_t \in S$.

$V = \{v_1, v_2, \dots, v_M\}$, l'ensemble discret des M symboles. On désigne un symbole au temps t par $o_t \in V$.

$A = \{a_{ij}\}_{1 \leq i,j \leq N}$, où $a_{ij} = P(q_{t+1} = s_j | q_t = s_i)$, pour le modèle d'ordre 1. A est la matrice des probabilités de transition entre états.

$B = \{b_j(k)\}_{1 \leq j \leq N, 1 \leq k \leq M}$, où $b_j(k) = P(o_t = v_k | q_t = s_j)$. B est la matrice des probabilités d'observation dans les états.

$\pi = \{\pi_i\}_{1 \leq i \leq N}$, où $\pi_i = P(q_1 = s_i)$. π est le vecteur des probabilités initiales des états.

Par simplification, on désignera un HMM par le triplet $\lambda = \{A, B, \pi\}$. Dans un HMM, les contraintes (markoviennes) suivantes doivent être respectées :

$$\sum_{i=1}^{N} \pi_i = 1,$$

$$\sum_{j=1}^{N} a_{ij} = 1, \qquad 1 \leq i \leq N,$$

$$\sum_{k=1}^{M} b_j(k) = 1, \qquad 1 \leq j \leq N$$

La figure 1 montre quelques exemples d'architectures possibles de HMMs en fonction des contraintes imposées sur les transitions entre états (transitions nulles) et sur les probabilités initiales. Ainsi, pour le modèle ergodique (voir figure 1.a), on a $\pi_i \neq 0, a_{ij} \neq 0$, $\forall i, j$ et pour les modèles gauche-droite (voir figure 1.b), $\pi_1 = 1, \pi_i = 0, a_{ii-1} = 0$, $j \leq i$ (pas de retour en arrière).

Les fonctionnalités d'un HMM sont l'évaluation de la probabilité d'observation d'une séquence O, l'apprentissage à partir d'un ensemble d'échantillons (séquences d'observations) et la reconnaissance d'une séquence. La description de ces fonctionnalités se base en grande partie sur l'article de Rabiner [Rab 89].

La probabilité d'observation de O sachant un modèle λ est la somme sur tous les chemins d'états Q des probabilités conjointes de O et de Q par rapport à ce modèle [Kri 90]. Une évaluation optimale de cette probabilité est obtenue par les fonctions *forward-backward* [Bau 67]. Nous allons définir la fonction *forward* $\alpha_t(i) = P(o_1 o_2 \dots o_t, q_t = s_i | \lambda)$ (la fonction backward β suit une définition duale) comme étant la probabilité des t premières observations sachant que le modèle se trouve dans l'état s_i :

1. Initialisation : $\alpha_1(j) = \pi_j b_j(o_1), \qquad 1 \leq j \leq N,$
2. Récursion : $\alpha_t(j) = \left[\sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij}\right] b_j(o_t), \; 1 \leq j \leq N,$ $t = 2 \dots T,$
3. Terminaison : $P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$

Le but de l'apprentissage est de déterminer les paramètres (A, B, π) qui maximisent le produit $\prod_{k=1}^{K} P(O^k|\lambda)$, où les O^k sont les séquences d'observations des échantillons d'apprentissage. Cela pose un problème relatif à l'absence de critère d'optimisation globale et de méthode directe. Les solutions utilisées ne présentent que des optimisations locales telles que les procédures d'estimation par MLE. Pour cette technique (qui est souvent employée), on utilise l'algorithme de Baum-Welch [Bau 68, Bau 72], basé sur le théorème de Baum qui garantit l'atteinte d'un maximum local de la fonction de vraisemblance par ré-estimation des paramètres A, B, π. Notons par $P^k = P(O^k|\lambda)$ la probabilité d'émission de l'échantillon O^k et par T^k sa longueur. A l'aide des fonctions α^k et β^k calculées pour chaque échantillon, nous définissons les quantités suivantes :

$$\xi_t^k(i,j) = P(q_t = s_i, q_{t+1} = s_j | O^k, \lambda) =$$
$$= \frac{1}{P^k} \alpha_t^k(i) b_j(o_{t+1}^k) \beta_{t+1}^k(j),$$

représentant la probabilité de transition entre s_i et s_j à l'instant t pour l'échantillon k.

$\zeta_t^k(i) = P(q_t = s_i|O^k, \lambda) = \sum_{j=1}^{N} \xi_t^k(i,j)$, étant la probabilité de se trouver dans l'état s_i à l'instant t pour l'échantillon k.

Les formules de ré-estimation des paramètres (A, B, π) s'écrivent à l'aide de ces quantités :

$$\overline{\pi}_i = \text{nb. de fois d'être dans } s_i \text{ à l'instant } (t=1) = \frac{1}{K}\sum_{k=1}^{K} \zeta_1^k(i),$$

$$\overline{a}_{ij} = \frac{\text{nb. transitions } s_i \to s_j}{\text{nb. transitions à partir de } s_i} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T^k-1} \xi_t^k(i,j)}{\sum_{k=1}^{K}\sum_{t=1}^{T^k-1} \zeta_t^k(i)},$$

$$\overline{b}_j(l) = \frac{\text{nb. de fois d'être dans } s_j \text{ et d'observer } v_l}{\text{nb. de fois d'être dans } s_j} = \frac{\sum_{k=1}^{K}\sum_{t=1 \text{ t.q. } o_t^k = v_l}^{T^k} \zeta_t^k(j)}{\sum_{k=1}^{K}\sum_{t=1}^{T^k} \zeta_t^k(j)}$$

La reconnaissance peut être effectuée de deux façons différentes : soit dans le cas d'un modèle par classe, par recherche du modèle discriminant (Model Discriminant), soit dans le cas d'un seul modèle pour toutes les classes, par recherche du chemin optimal qui fournira la classe (Path Discriminant) [Che 93a].

Dans le premier cas, la reconnaissance peut se faire simplement par le calcul des probabilités d'émission de la forme par les modèles que l'on suppose *a priori* équiprobables. La forme à reconnaître est affectée à la classe dont le modèle fournit la probabilité la plus importante :

$$\lambda^* = \underset{\lambda \in \Lambda}{\operatorname{argmax}} P(O|\lambda) \qquad (3)$$

où Λ désigne l'ensemble des modèles.

Dans le deuxième cas, la reconnaissance consiste à déterminer le chemin correspondant à l'observation, c'est-à-dire à trouver dans le modèle, la meilleure suite d'états, appelée *suite d'états de Viterbi*, qui maximise la quantité $P(Q|O,\lambda)$. Ceci revient à trouver le meilleur chemin dans un graphe. La structure de ce graphe se prête aux techniques de la programmation dynamique. Pour cela, l'algorithme de Viterbi [For 73] définit $\delta_t(i)$ qui est la probabilité du meilleur chemin amenant à l'état s_i à l'instant t, en étant guidé par les t premières observations :

$$\delta_t(i) = \max_{q_1,\ldots,q_{t-1}} P(q_1, q_2, \ldots, q_t = s_i, o_1, o_2, \ldots, o_t|\lambda) \qquad (4)$$

c'est-à-dire que $\delta_t(i)$ est la meilleure correspondance entre la suite $q_1 q_2 \ldots q_t$ et la suite $o_1 o_2 \ldots o_t$ avec la contrainte $q_t = s_i$. Par induction, on calcule :

1. Initialisation : $\delta_1(j) = \pi_j b_j(o_1), \qquad 1 \leq j \leq N,$
2. Récurrence croissante : $\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t),$

   $\gamma_t(s_j) = \underset{s_i \in S}{\operatorname{argmax}} [\delta_{t-1}(i) a_{ij}], \quad 1 \leq j \leq N,\ t = 2 \ldots T,$
3. Terminaison : $P^*(O|\lambda) = \max_{1 \leq i \leq N} \delta_T(i),$
4. Séquence d'états $Q^* = \{q_t^*\}_{1 \leq t \leq T}$ obtenue par récurrence décroissante : $q_{t-1}^* = \gamma_t(q_t^*), \qquad t = T \ldots 2$

On garde trace, lors du calcul, de la suite d'états qui donne le meilleur chemin amenant à l'état s_i à l'instant t.

2.3. décomposition probabiliste

Le cadre de la reconnaissance par les modèles de Markov est le domaine des probabilités. Dans ce domaine, on reconnaît une forme en lui associant une étiquette qui maximise la probabilité conditionnelle de cette étiquette sachant la description de la forme (probabilité *a posteriori* de l'étiquette).

Par ailleurs, un modèle probabiliste qui intègre plusieurs échantillons par apprentissage est capable de synthétiser des probabilités d'affectation de nouvelles formes, ce qui revient à dire que l'on a la probabilité conditionnelle de la forme sachant le modèle (vraisemblance de la forme).

On utilise la règle de Bayes qui met en équation ces deux types de probabilités. Elle fait de plus intervenir la probabilité *a priori* de la forme et du modèle.

Quand les formes sont décomposables en sous-formes, on souhaite décomposer les probabilités générales (de la forme et/ou de l'étiquette) et de ramener le problème de la maximisation des probabilités au niveau de ces sous-formes. Cette décomposition s'obtient en faisant des hypothèses sur la dépendance entre sous-formes elles-mêmes, entre sous-formes et étiquettes ou entre étiquettes elles-mêmes. Ces hypothèses ne sont pas générales mais dépendent de l'application traitée. Certaines hypothèses font aboutir à des décompositions des probabilités simulables par des modèles de Markov.

Nous allons rappeler dans la suite la formule de Bayes et étudier quelques cas de décomposition conduisant à une modélisation markovienne.

Soit F une forme à reconnaître et E une étiquette possible de F. La règle de Bayes nous donne la probabilité *a posteriori* de E comme suit :

$$P(E|F) = \frac{P(F|E)P(E)}{P(F)} \propto P(F|E)P(E) \qquad (5)$$

Si l'on considère F comme une séquence de sous-formes élémentaires $F = f_1 \ldots f_n$ et $E = e_1 \ldots e_n$, plusieurs hypothèses

d'indépendance sur les termes $P(F|E)$ et $P(E)$ peuvent être formulées. Pour toute interprétation ultérieure de l'équation (5), F représente la description de la forme et E, celle de l'étiquette. Dans le cas précis d'une modélisation de type HMM, F peut être assimilée à une séquence d'observations (notation O).

2.3.1. décomposition de la forme par rapport à l'étiquette

Ceci correspond au cas où il existe une relation biunivoque entre les sous-formes et leurs étiquettes et lorsque celles-ci sont supposées indépendantes (ce qui signifie que la vraisemblance d'une sous-forme ne dépend que de l'étiquette de cette sous-forme). Dans ce cas, la vraisemblance de la forme se traduit par un produit de probabilités conditionnelles élémentaires.

$$P(F|E) = \prod_{i=1}^{n} P(f_i|e_i) \tag{6}$$

Nous verrons dans la suite qu'en combinant les termes $P(f_i|e_i)$ avec des termes issus d'une décomposition appropriée de la probabilité *a priori* $P(E)$ de l'étiquette, cela peut conduire à une modélisation markovienne d'un ordre donné.

Exemple 1 : correction lexicale de mots

Dans ce cas, il s'agit de vérifier la cohérence lexicale d'un mot à partir des résultats de la reconnaissance de ses caractères.

Soit $(f_1, \ldots, f_n)$ la suite de caractères reconnus. Il s'agit de trouver la séquence d'étiquettes $(e_1, \ldots, e_n)$ qui maximise la probabilité *a posteriori* $P(e_1, \ldots, e_n|f_1, \ldots, f_n)$. Si l'on suppose que la langue est une source de Markov [Neu75a] et si l'on considère que les caractères ont été reconnus de manière indépendante les uns des autres, dans ce cas l'attention sera portée sur :

1. la modélisation de la source : soit e_k le $k$ème caractère du texte. En considérant que le texte suit une chaîne de Markov du premier ordre, on a :

$$P(e_k|e_1, \ldots, e_{k-1}) = P(e_k|e_{k-1}) \tag{7}$$

où les quantités $P(e_k|e_{k-1})$ (bigrammes) peuvent être estimées par comptage à partir d'un dictionnaire.

2. le module de reconnaissance de caractères (OCR) : pour simplifier, quelques considérations sont prises concernant :

(a) la synchronisation avec le texte entraînant le fait que chaque caractère du texte e_k est lu dans le bon ordre et produit un seul caractère f_k.

(b) le module de reconnaissance n'a pas de mémoire, c'est-à-dire :

$$P(f_k|e_1, \ldots, e_n) = P(f_k|e_k) \tag{8}$$

A l'aide de ces hypothèses, on peut écrire :

$$P(f_1, \ldots, f_n|e_1, \ldots, e_n) = \prod_{i=1}^{n} P(f_i|e_i) \tag{9}$$

exprimant la probabilité de reconnaissance ou de confusion d'une forme avec une étiquette. On peut également écrire à partir de l'hypothèse (7), avec la convention $P(e_1|e_0) = P(e_1)$, la quantité :

$$P(e_1, \ldots, e_n) = \prod_{i=1}^{n} P(e_i|e_{i-1}) \tag{10}$$

qui indique que la probabilité *a priori* du texte se ramène à un produit de probabilités de succession de couples de caractères (bigrammes).

Trouver la séquence $e_1, \ldots, e_n$ revient à maximiser :

$$P(f_1, \ldots, f_n|e_1, \ldots, e_n)P(e_1, \ldots, e_n) = \prod_{i=1}^{n} P(f_i|e_i)P(e_i|e_{i-1}) \tag{11}$$

L'algorithme de Viterbi classique essaie de trouver, de manière efficace dans ce modèle, la meilleure suite de caractères possibles maximisant la vraisemblance pour un mot donné, en utilisant le principe de la programmation dynamique. Cette comparaison dynamique par rapport à toutes les suites possibles conduit à un calcul très lourd. En effet, dans l'algorithme classique de Viterbi, il y a pour chaque position de lettre dans un mot, autant d'alternatives que les lettres de l'alphabet [Tou 78, Hul 83]. Or, nous savons que dans certaines langues, certaines lettres ne se suivent pas. En outre, l'OCR est là pour limiter les choix possibles pour chaque position. Toussaint propose un algorithme de Viterbi modifié (MVA) qui, tout en gardant la même optimalité que l'algorithme classique, autorise des profondeurs de recherche variables [Shi 79]. Shinghal et Toussaint [Shi 79] affirment que, pour l'anglais, la profondeur optimale est 2. Hull *et al.* [Hul 83] limitent leur recherche à l'aide d'un lexique.

Dans nos travaux sur la reconnaissance de textes imprimés multi-fontes [Bel94a, Ani95a], il nous a paru judicieux de faire intervenir le taux de confiance de l'OCR directement dans l'algorithme de Viterbi modifié. La profondeur maximale est de 3, correspondant aux premières réponses de l'OCR. Dans certains cas, la profondeur est plus petite (par exemple, l'OCR donne une seule réponse) d'où une efficacité de recherche accrue de la chaîne.

Les modèles sont de type séquentiel à branches parallèles, construits séparément pour les mots de même longueur d'un lexique de 190 000 mots (24 modèles pour le français). De cette manière, les probabilités de transition (bigrammes) sont dépendantes de la longueur des mots.

La figure 2 montre un exemple de mot à sept lettres pour les sorties suivantes de l'OCR : **(m,w), a, (l,t,f), a, i, (s,z), e** où le contenu des parenthèses donne les alternatives de l'OCR. Les

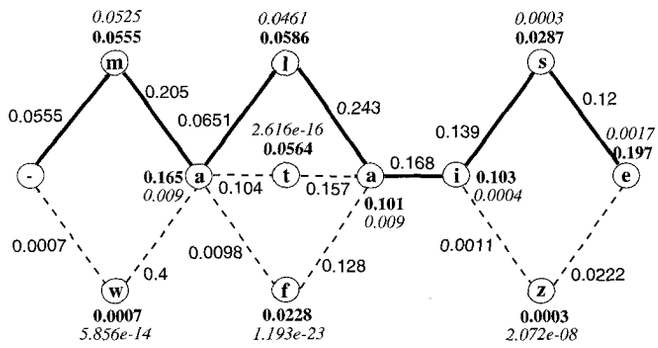


Figure 2. – Réseau de Viterbi pondéré pour une chaîne de 7 lettres.

scores de reconnaissance de l'OCR sont montrés en italique. Le chemin suivi par l'algorithme de reconnaissance est en gras.

Nous avons effectué de nombreux tests afin de comparer les différentes méthodes de correction contextuelle basées sur des HMMs (d'ordre 1 avec confiance de l'OCR et d'ordre 2). Nous avons constaté que pour certaines longueurs de mots où il n'y avait pas assez d'échantillons pour construire les trigrammes, le bigramme renforcé donne de meilleurs résultats. Notre conclusion est que, en l'absence d'un correcteur lexical, le HMM du premier ordre ayant pris en compte des scores de confiance de l'OCR est la meilleure méthode.

D'autres solutions combinant les réponses de l'OCR et des dictionnaires ont été également proposées [Shi 79, Hul 83, Sri 84].

Exemple 2 : Estimation de la plausibilité d'une phrase

On se restreint dans cet exemple à l'estimation de la plausibilité à partir des probabilités de succession de groupes de mots et des probabilités de reconnaissance de ces mots. On peut trouver dans la littérature, plusieurs exemples de reconnaissance stochastique de phrases qui font appel à ce type d'estimation [Bau 72, Bah 83, Gil 92c].

Dans l'exemple des montants littéraux de chèques postaux [Gil 92 c], il s'agit d'identifier les mots composants et de vérifier la cohérence syntaxique de l'ensemble. Soit E la phrase étiquette correspondant à l'image F du montant (les e_i sont les étiquettes des mots f_i). En faisant l'hypothèse que la vraisemblance d'une image f_i d'un mot ne dépend que de l'étiquette associée e_i, la vraisemblance du montant peut se décomposer selon l'équation (6).

En supposant que le montant suit un processus de Markov d'ordre 2, la probabilité *a priori* du montant s'écrit :

$$P(E) = P(e_1, \ldots, e_n) = \\ = P(e_1)P(e_2|e_1)\prod_{i=3}^{n} P(e_i|e_{i-1}, e_{i-2}) \text{ (trigrammes)} \quad (12)$$

Dans ce cas, l'hypothèse d'indépendance indique que la vraisemblance de chaque image de mot du montant ne dépend que de l'étiquette associée. En choisissant un processus d'ordre 2, on peut gérer des dépendances lointaines entre mots. Par exemple, une succession « ...dix neuf cent... » n'a pas de sens mais ceci reste indécelable par un processus d'ordre 1. Un autre avantage de l'utilisation des trigrammes est une prise en compte plus fine des régularités relatives à l'application comme le fait que les montants arrondis sont plus fréquents (et donc plus probables). Ce système dissocie les niveaux mot et phrase (montant), par opposition aux systèmes de correction lexicale précédents et fonctionne de manière complètement ascendante, n'étant pas guidé par la syntaxe de la phrase.

Un autre exemple vise la reconnaissance de codes postaux dans les adresses [Sri 93, Gil 93, Coh 94]. Un code postal est composé d'une suite de chiffres dont le nombre varie suivant les pays. Si l'on fait l'hypothèse que chaque chiffre peut être identifié séparément des autres (les chiffres sont séparés pendant une phase préalable de segmentation et la reconnaissance d'un chiffre n'influe pas sur la reconnaissance d'un autre chiffre), la probabilité conditionnelle peut s'écrire comme dans (6) où les f_i sont ici les images des chiffres et les e_i, les chiffres correspondants. La probabilité *a priori* du code $P(e_1 \ldots e_n)$ dépend de certains critères relatifs, en France par exemple, à la démographie des départements ou des grandes villes.

2.3.2. décomposition par association à un modèle

Dans ce cas, la forme F est décomposable en sous-formes et il existe un modèle λ_E associé à l'étiquette E. La probabilité *a posteriori* de l'étiquette E devient celle du modèle, d'après l'équation (5) :

$$P(E|F) = P(\lambda_E|F) \propto P(F|\lambda_E)P(\lambda_E) \quad (13)$$

où $P(\lambda_E)$ est la probabilité *a priori* du modèle λ_E, pouvant être estimée pendant l'apprentissage, par le rapport entre le nombre d'échantillons ayant servi à la construction du modèle λ_E et le nombre de tous les échantillons de l'apprentissage.

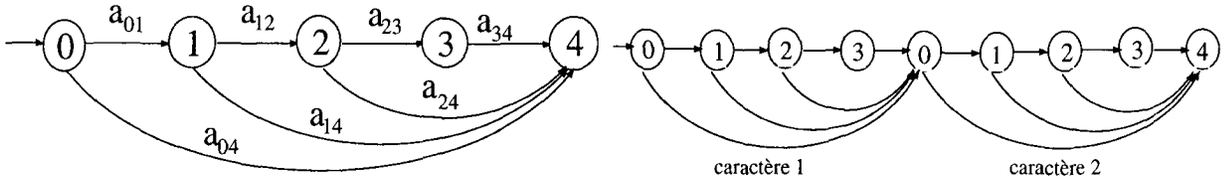
L'idée est d'associer à la forme une suite d'états du modèle pour observer ces sous-formes.

Dans le domaine des probabilités conditionnelles, le terme $P(F|\lambda_E)$ se décompose en une somme sur toutes les séquences d'états du modèle de longueur n, du produit de la probabilité conditionnelle de la forme sachant la séquence d'états et de la probabilité de la séquence sachant le modèle.

$$P(F|\lambda_E) = P(f_1 \ldots f_n|\lambda_E) = \\ = \sum_{q_1,\ldots,q_n} P(f_1, \ldots, f_n|q_1, \ldots, q_n, \lambda_E)P(q_1, \ldots, q_n|\lambda_E) \quad (14)$$

On suppose souvent qu'un seul chemin contribue significativement au calcul de $P(F|\lambda_E)$ que l'on va dénoter, par la suite, par $\{q_i^*\}$. L'équation précédente devient :

$$P(F|\lambda_E) = P(f_1 \ldots f_n|\lambda_E) = \\ = P(f_1, \ldots, f_n|q_1^*, \ldots, q_n^*, \lambda_E)P(q_1^*, \ldots, q_n^*|\lambda_E) \quad (15)$$



(a) Explicitation de la notion de durée

(b) Modèle de mot

Figure 3. – Modèles de caractère et de mot, d'après Chen et Kundu.

$\{q_i^*\}$ peut être généralement modélisé comme un processus stochastique stationnaire d'ordre 1, ce qui donne :

$$P(q_1^*, \ldots, q_n^* | \lambda_E) = P(q_1^* | \lambda_E) \prod_{i=2}^{n} P(q_i^* | q_{i-1}^*, \lambda_E) \qquad (16)$$

La quantité $P(f_1, \ldots, f_n | q_1^*, \ldots, q_n^*, \lambda_E)$ peut connaître différents développements en fonction des hypothèses de dépendance que l'on peut faire entre les sous-formes et les états d'observation. Nous allons donner dans la suite quelques exemples d'hypothèses.

Cas 1 : l'observation de la forme ne dépend que de l'état associé :

$$P(f_1, \ldots, f_n | q_1^*, \ldots, q_n^*, \lambda_E) = \prod_{i=1}^{n} P(f_i | q_i^*, \lambda_E) \qquad (17)$$

Cette formulation correspond à l'utilisation classique des HMMs où les $P(f_i | q_i^*, \lambda_E)$ représentent les distributions de probabilités d'observation. La probabilité d'observation peut être évaluée de différentes manières. Dans ce cas précis où l'observation ne dépend que de l'état associé, l'évaluation de la distribution dans l'état peut se faire soit de manière discrète, soit de manière continue.

Les HMMs qui présentent des distributions continues sont connus sous le nom de modèles semi-continus. De tels modèles ont été utilisés par Chen *et al.* [Che 93b] pour la reconnaissance de mots manuscrits correspondant à des noms de villes dans des adresses postales. Après une procédure de pré-segmentation du mot en graphèmes, les auteurs extraient pour chaque segment 35 primitives (géométriques et topologiques) à variation continue. Dans chaque état du modèle, l'observation est constituée d'un vecteur de telles primitives. A cause de la multiplicité des styles d'écriture, on construit par analyse de ces vecteurs un mélange de gaussiennes permettant d'estimer chacune une distribution de probabilités continue relative à l'observation d'un style d'écriture particulier. Pour chaque état j, on identifie par *k-means*, à partir d'échantillons d'apprentissage, M_j styles d'écriture. Ensuite, on estime la probabilité d'observation $b_j(x)$ par :

$$\begin{aligned} b_j(x) &= P(f_t = x | q_t = s_j, \lambda_E) = \\ &= \sum_{m=1}^{M_j} c_{jm} . \mathcal{N}[x, \mu_{jm}, U_{jm}], 1 \leq j \leq N \end{aligned} \qquad (18)$$

où la $m^{ème}$ composante du mélange dans l'état j suit une distribution normale $\mathcal{N}$ de moyenne μ_{jm} et de matrice de covariance U_{jm} (supposée diagonale), x étant le vecteur modélisé.

Ce système montre bien que la prise en compte du style est un facteur important dans la modélisation de l'écriture mais la continuité non assurée d'un état à l'autre du HMM diminue de son intérêt. Une recherche préalable du style à l'aide de primitives adéquates serait plus judicieuse comme cela a été montré par Crettez dans [Cre 94, Cre 96].

Une autre manière de faire dépendre l'observation de l'état est de quantifier sa durée dans l'état [Lev 83, Rab 89]. Ceci permet d'associer un sens à l'analyse d'une entité physique dans un état (caractère, mot, etc.). Dans Chen et Kundu [Che 93a, Che 93b], la durée discrète correspond au nombre de segments d'écriture (graphèmes) possibles pouvant être extraits par segmentation de chaque caractère (maximum 4). La figure 3.a montre comment le modèle simule les différentes possibilités de segmentation d'un caractère en graphèmes. La durée est explicitée par les différentes transitions entre états. La reconnaissance du mot utilise l'algorithme MVA avec incorporation de la durée dont la récursion est :

$$\delta_t(j) = \max_{1 \leq i \leq N} \max_{1 \leq d \leq 4} \delta_{t-d}(i) a_{ij} P(d | q_j) b_j(f_{t-d+1}^t) \qquad (19)$$

où f_{t-d+1}^t représente le vecteur des primitives extraites de la concaténation des graphèmes $t - d + 1 \ldots t$. Finalement, les auteurs signalent un taux de reconnaissance de mots de 70.2% pour un vocabulaire de 271 noms de villes.

On retrouve la même idée d'utilisation de la durée explicite dans [Aga 93] pour la reconnaissance de textes imprimés. Contrairement au système précédent, la durée est estimée de manière continue comme sera montré dans 3.2.

Dans le cas où les probabilités d'observation dans les états sont discrètes, on utilise classiquement l'algorithme de Baum-Welch [Bau 72] pour les estimer. La plupart des systèmes de RAE utilisant les HMMs peuvent être rangés dans cette catégorie [Boz 89, Ani 91, Gil 92a, Aga 93, Bos 94, Kuo 94].

Un exemple d'un tel système est celui de Gillies [Gil 92a], utilisé pour la reconnaissance de mots dans les adresses postales. Pour chaque mot du vocabulaire, il construit un modèle de type gauche-droite, par concaténation de modèles de lettres qui sont

également de type gauche-droite, mais sans saut d'état. L'idée générale de ces modèles est que la distribution des probabilités des symboles observés dans des zones de la lettre, varie selon la zone en traversant la lettre de gauche à droite. Dans chaque zone, il extrait des primitives topologiques et géométriques. Ensuite, il constitue par quantification vectorielle un ensemble de prototypes. A ces prototypes est associé un vocabulaire de symboles utilisés par les modèles de Markov. Ainsi, l'image est transformée en une séquence de symboles. Les sous-séquences de symboles sont délimitées et servent à l'apprentissage des modèles lettres correspondants. Le taux de reconnaissance varie de 72.6% (première réponse) à 90.5% (10 premières réponses) pour un vocabulaire de 100 mots et de 51.0% à 80.1% pour un lexique de 1000 mots.

La notion de concaténation de modèles de lettres est souvent employée [Gil 92b, Che 92, Che 93a, Ber 93] et peut être exprimée avec les notations introduites auparavant comme suit :

$$P(f_i|q_i = s_j, \lambda_E) = P(f_i|\lambda_{e_j}), \qquad i = 1 \ldots n \tag{20}$$

où $E = e_1 \ldots e_n$ et $\lambda_E = \lambda_{e_1} \oplus \ldots \oplus \lambda_{e_n}$, avec $\oplus$ désignant la concaténation de deux HMMs.

La concaténation de modèles ou la sémantique de l'opérateur $\oplus$ diffère selon les auteurs. Par exemple, dans le cas de Gillies, une transition simple est ajoutée pour faire le lien entre deux modèles consécutifs, tandis que dans Bercu [Ber 93] ou dans Chen [Che 93a], le dernier état du modèle est le premier du suivant.

Cas 2 : l'observation de la forme dépend à la fois de l'état courant et de l'état précédent, dans lequel se trouvait le modèle :

$$P(f_1 \ldots f_n|q_1^* \ldots q_n^*, \lambda_E) = p(f_1|q_1^*) \prod_{i=2}^{n} p(f_i|q_i^*, q_{i-1}^*, \lambda_E) \tag{21}$$

Dans le modèle de Bercu *et al.* [Ber 92, Ber 93], la distribution des probabilités d'observation des primitives dépend de l'état courant et de l'état précédent (voir figure 4). Cette observation est associée aux transitions du modèle :

$$P(f_t = x|q_t = s_j, q_{t-1} = s_i) = b_{ij}(x)$$

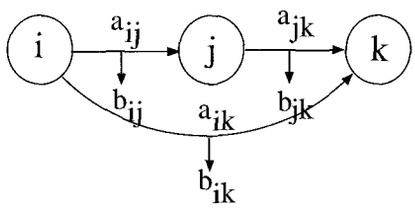


Figure 4. – Modèle de lettre, d'après Bercu et Delyon.

Le système général utilise un HMM à deux niveaux pour la reconnaissance en ligne de mots : un niveau local décrivant les primitives dans les lettres (boucles, pics et arcs orientés) et un niveau global d'observation des lettres dans le mot (extensions par rapport à la bande centrale). Le HMM est décrit par un triple processus stochastique; une chaîne de Markov correspondant à la suite des états Q, un processus stochastique associé à l'observation locale O et un autre associé à l'observation globale G. Les problèmes liés à la reconnaissance et à l'apprentissage reviennent à déterminer le chemin optimal estimé par MAP :

$$Q^* = \underset{Q}{\operatorname{argmax}}\, P(Q|O, G) = \underset{Q}{\operatorname{argmax}}\, P(Q, O, G) \tag{22}$$

Pour évaluer la probabilité conjointe du chemin d'états, de la séquence locale et de la séquence globale d'observations, les auteurs proposent l'algorithme de Viterbi suivant. Notons par :

$$\delta_t(i) = \max_{q_1,\ldots,q_t} P(q_1, \ldots, q_{t-1}, q_t = s_i, o_1, \ldots, o_t, g_1, \ldots, g_t) \tag{23}$$

la probabilité du chemin optimal partiel amenant à s_i, en étant guidé par les t premières observations (locales et globales). δ suit la récurrence suivante :

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij} b_{ij}(o_t)] \tag{24}$$

si s_j est un état intermédiaire d'un modèle de lettre et :

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij} P(g_{t_k}, \ldots, g_{t_{k+1}-1}|e_k)] \tag{25}$$

si s_j est l'état final pour la lettre e_k. t_k, t_{k+1} sont les limites des sous-séquences d'observations correspondant à e_k. Notons encore que la quantité $P(g_{t_k}, \ldots, g_{t_{k+1}-1}|e_k)$ représente la vraisemblance des observations globales pour e_k et peut être apprise.

L'avantage de ce système se situe dans l'intégration de deux niveaux complémentaires d'information : un niveau de description locale des lettres (boucles, pics et bosses) et un niveau global au niveau du mot (hampes, jambages) permettant de discriminer entre lettres ayant la même description locale. Cependant, ce système nécessite une extraction préalable de primitives de haut niveau et donc des prises de décision avant la reconnaissance. Par ailleurs, la pertinence de cette modélisation dépend de la complémentarité entre les différents niveaux de primitives qui reste à prouver.

Cas 3 : Cas où la probabilité d'observation d'une sous-forme dépend de l'état courant et de la sous-forme précédente :

$$P(f_1 \ldots f_n|q_1^* \ldots q_n^*, \lambda_E) = p(f_1|q_1^*) \prod_{i=2}^{n} P(f_i|q_i^*, f_{i-1}, \lambda_E) \tag{26}$$

Beaucoup de systèmes en RAE utilisent une pré-segmentation en graphèmes avant la reconnaissance. Cette pré-segmentation peut parfois couper les lettres en parties. Si l'on veut faire un apprentissage par lettre, la pré-segmentation pose un problème réel de choix d'échantillons. Lemarié [Lem 93, Lem 96] utilise un réseau de neurones (Radial Basis Function ou RBF) entraîné à partir de lettres et de segments de lettres pour évaluer les densités $P(f_i|q_i^*, f_{i-1})$. Ce système permet de lier deux segments consécutifs pour estimer la présence potentielle d'une lettre.

L'entrée du réseau (RBF) utilisé est un couple de deux segments consécutifs f_i, f_{i-1} pouvant appartenir à une même lettre ou à deux lettres consécutives. La sortie est l'état le plus probable du HMM. En résumant, le réseau est capable d'estimer $P(q_i^*|f_i, f_{i-1})$. Cette quantité est introduite par la règle de Bayes appliquée localement :

$$P(f_i|q_i^*, f_{i-1}) = \frac{P(q_i^*|f_i, f_{i-1})P(f_i|f_{i-1})}{P(q_i^*|f_{i-1})} \qquad (27)$$

Notons encore que $P(f_i|f_{i-1})$ est constante et que $P(q_i^*|f_{i-1})$ est également évaluée par un réseau RBF.

Ce modèle hybride associant un réseau neuronal RBF et un HMM semble être très efficace pour résoudre les problèmes de sursegmentation de lettres. En effet, la composante HMM trouve l'alignement des lettres du mot sur les parties d'image en étant guidée par les probabilités d'observation fournies par le réseau dans les différents états. Ces probabilités sont estimées de manière très fine par la prise en compte du contexte gauche représenté par le segment d'image précédent. Cependant, l'apprentissage reste empirique car il ne bénéficie pas d'une formalisation adaptée et sa convergence n'est pas facile à démontrer.

3. modèles de Markov pseudo-2D planaires

Les HMMs, en tant qu'outils statistiques, permettent de calculer la probabilité d'appartenance d'une forme à une classe en fonction du degré de distorsion subi par cette forme. Pour les signaux unidimensionnels, comme la parole, la distorsion, qui est essentiellement de type contraction et dilatation temporelle, est calculée à l'aide de l'algorithme DTW (dynamic time warping). Cet algorithme effectue un appariement élastique entre un prototype et une forme. Dans le HMM, cet appariement est fait de manière optimale par l'algorithme de Viterbi. Le prototype est synthétisé par apprentissage à partir de plusieurs échantillons.

L'utilisation de ces HMMs en RAE a permis d'obtenir des résultats intéressants pour certaines applications. Mais la nature 2D de l'écriture (différence fondamentale avec la parole) permet de penser que des améliorations plus importantes peuvent être apportées en étendant les HMMs à deux dimensions. Cependant, il a été démontré [Lev 92] qu'une extension directe de l'algorithme DTW, appelée DPW (dynamic plane warping), conduit à un problème NP complet. Néanmoins, en appliquant certaines contraintes à l'appariement, on peut ramener le problème à une complexité polynomiale.

Le but du DPW est d'apparier une image de référence avec une image de test via une fonction de mise en correspondance de manière à ce que la distorsion soit minimale. La fonction doit satisfaire des conditions globales sur les limites des images et doit être localement monotone en x et y. Si on impose en plus la séparabilité de la fonction en ses variables, ce qui revient à dire que les distorsions horizontales sont indépendantes de celles verticales, on peut alors définir les PHMMs (planar ou pseudo-HMM).

3.1. définition des PHMMs

Les PHMMs sont des HMMs où la probabilité d'observation dans chaque état est donnée par un HMM secondaire [Gil 94]. On fera la distinction entre le modèle principal composé de super-états et les modèles secondaires associés aux super-états. Pour une image, le modèle principal fera l'analyse selon une direction (par exemple, la direction verticale) et les modèles secondaires la feront selon l'autre axe. En RAE, conformément au sens de l'écriture, les modèles secondaires sont souvent associés aux lignes où la forme est réellement observée et leurs architectures sont typiquement gauche-droite (voir figure 5). Le modèle principal, orienté verticalement, fera la corrélation (dépendance probabiliste) de ces observations d'une manière globale. Généralement, on associe plusieurs lignes à chaque super-état, pensant que plusieurs lignes consécutives sont étroitement corrélées entre elles et donc, peuvent être analysées par un même HMM. Le nombre de super-états dépend de la morphologie de la forme et des principales zones horizontales d'observation que l'on veut mettre en évidence. On donnera plus loin quelques exemples d'architectures de PHMMs dédiées à l'écriture.

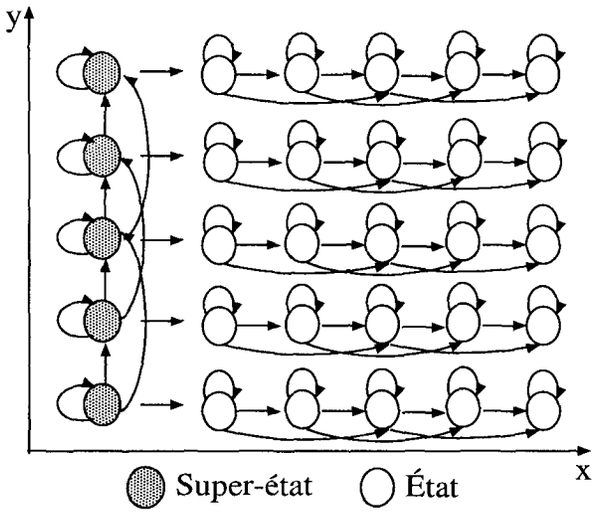


Figure 5. – Exemple d'architecture d'un PHMM.

En faisant appel à des extensions des notations de [Rab 89], on peut définir un PHMM de la manière suivante :

Définition 7

Un PHMM est donné par le triplet $\eta = (A, \pi, \Lambda)$:

$S = \{s_1, s_2, \ldots, s_N\}$, l'ensemble des N super-états du modèle. On désigne un super-état pour la ligne y par $q_y \in S$.

$V = \{v_1, v_2, \ldots, v_M\}$, l'ensemble discret des M symboles. On désigne un symbole à la position (x, y) par $o_{xy} \in V$.

$A = \{a_{ij}\}_{1 \leq i,j \leq N}$, où $a_{ij} = P(q_{y+1} = s_j | q_y = s_i)$, la matrice de transition entre super-états.

$\pi = \{\pi_i\}_{1 \leq i \leq N}$, où $\pi_i = P(q_1 = s_i)$ est la probabilité initiale du super-état s_i.

$\Lambda = \{\lambda^k\}_{1 \leq k \leq N}$, l'ensemble des HMMs associés aux super-états. $\lambda^k = (A^k, B^k, \pi^k)$, où :

$S^k = \{s_1^k, s_2^k, \ldots, s_{N^k}^k\}$, l'ensemble des N^k états du modèle, l'état localisé en (x, y) étant noté par $q_{xy} \in S^k$. $A^k = \{a_{ij}^k\}_{1 \leq i,j \leq N^k}$, où $a_{ij}^k = P(q_{x+1y} = s_j^k | q_{xy} = s_i^k)$, la matrice de transition entre états.

$B^k = \{b_j^k(l)\}_{1 \leq j \leq N^k, 1 \leq l \leq M}$,

où $b_j^k(l) = P(o_{xy} = v_l | q_{xy} = s_j^k)$

$\pi^k = \{\pi_i^k\}_{1 \leq i \leq N^k}$, où $\pi_i^k = P(q_{1y} = s_i^k)$.

La reconnaissance se fait en évaluant la mesure de distorsion bayésienne P^* entre le mot échantillon et le modèle η. Se donnant un modèle η et une observation O, il s'agit de trouver la séquence d'états $Q = \{q_1, \ldots, q_Y\}$ qui maximise $P(Q|O, \eta)$.

On définit $\Delta_y(j)$ comme étant la probabilité la plus forte le long d'un chemin dans la ligne y :

$$\Delta_y(j) = \max_{q_1, \ldots, q_{y-1}} P(q_1 \ldots q_y = s_j, o_1 \ldots o_y | \eta) \qquad (28)$$

Le calcul de (28) implique le calcul de $P_j(y)$ qui est la probabilité de la ligne y dans le super-état j, c'est-à-dire : $P_j(y) = P(o_y | q_y = s_j)$, obtenue par une autre exécution de l'algorithme de Viterbi. Ainsi, la procédure complète de mesure de distorsion d'un mot avec un PHMM est un algorithme de Viterbi doublement intégré, fonctionnant comme suit :

– Initialisation

$$\Delta_1(j) = \pi_j P_j(1), \qquad 1 \leq j \leq N \qquad (29)$$

– Récursion

$$\Delta_y(j) = \max_{1 \leq k \leq N} [\Delta_{y-1}(k) a_{kj}] P_j(y), \quad 2 \leq y \leq Y, \; 1 \leq j \leq N \qquad (30)$$

$$\Gamma_y(s_j) = \operatorname*{argmax}_{s_k \in S} [\Delta_{y-1}(k) a_{kj}], \quad 2 \leq y \leq Y, \; 1 \leq j \leq N \qquad (31)$$

– Terminaison

$$P^* = \max_{1 \leq j \leq N} [\Delta_Y(j)] \qquad (32)$$

– Séquence de super-états $Q^* = \{q_y^*\}_{1 \leq y \leq Y}$

$$q_{y-1}^* = \Gamma_y(q_y^*), \qquad y = Y \ldots 2 \qquad (33)$$

Le calcul de $P_j(y)$ s'obtient à partir d'un modèle 1D par une autre exécution de l'algorithme de Viterbi :

– Initialisation

$$\delta_{1y}^j(i) = \pi_i^j b_i^j(o_{1y}), \qquad 1 \leq i \leq N^j \qquad (34)$$

– Récursion

$$\delta_{xy}^j(i) = \max_{1 \leq k \leq N^j} [\delta_{x-1,y}^j(k) a_{ki}^j] b_i^j(o_{xy}), \quad 2 \leq x \leq X, \; 1 \leq i \leq N^j \qquad (35)$$

$$\gamma_{xy}^j(s_i^j) = \operatorname{argmax}_{s_k^j \in S^j} [\delta_{x-1,y}^j(k) a_{ki}^j], \quad 2 \leq x \leq X, \; 1 \leq i \leq N^j \qquad (36)$$

– Terminaison

$$P_j(y) = \max_{1 \leq i \leq N^j} [\delta_{Xy}^j(i)] \qquad (37)$$

– Séquence d'états $\{q_{xy}^*\}_{1 \leq x \leq X}$

$$q_{x-1,y}^* = \gamma_{xy}^j(q_{xy}^*), \qquad x = X \ldots 2 \qquad (38)$$

3.2. application à la RAE

Les PHMMs ont été appliqués récemment sur l'imprimé par [Lev 92, Aga 93, Kuo 94, Bos 94, Ben 96] et sur le manuscrit hors-ligne par [Gil 94].

Pour l'imprimé, Agazzi et Kuo [Kuo 94] proposent une architecture PHMM pour les caractères et les mots, comprenant un modèle vertical de super-états et des modèles horizontaux, un par super-état, comme le montre la figure 6 pour un caractère isolé. Le nombre d'états et de super-états est déterminé manuellement, en fonction de la topologie de la forme étudiée, c'est-à-dire de l'existence de zones informatives.

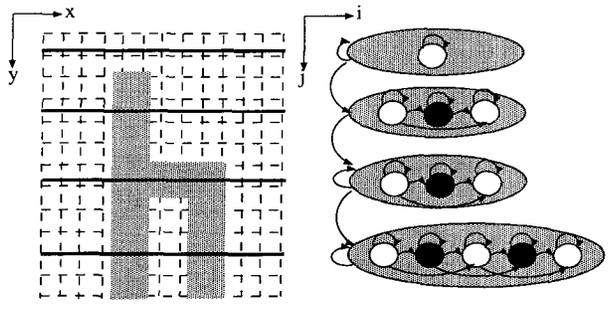


Figure 6. – PHMM pour les caractères imprimés, d'après Agazzi et Kuo.

L'observation est réalisée directement sur la matrice de pixels et est représentée par un vecteur de quatre composantes, correspondant à la valeur du pixel (0 ou 1), à son voisinage horizontal codé binairement (0 à 7), et aux deux positions relatives du pixel respectivement dans sa colonne et sa ligne (valeur parmi 0 ... 10 (tranches réalisées)).

Pour l'apprentissage, les paramètres du modèle, A, π et Λ sont estimés en utilisant l'algorithme *k-means* qui permet de faire des regroupements des observations en classes correspondant aux zones des super-états. La séquence d'observations de chaque échantillon est segmentée en une séquence de super-états, en déterminant l'alignement optimal de l'échantillon avec le modèle courant, à l'aide de l'algorithme de Viterbi. Les paramètres du modèle sont ensuite ré-estimés, par comptage, en fonction des histogrammes des résultats de la segmentation. En cas de non convergence, le lot d'apprentissage est réutilisé et les paramètres sont de nouveau ré-estimés. Les paramètres initiaux sont donnés manuellement et de manière arbitraire.

La probabilité d'observation b_i^j d'un vecteur o_{xy} dans l'état s_i^j est exprimée par :

$$b_i^j(o_{xy}) = \prod_{m=1}^{m=4} b_{im}^j[o_{xy}(m)]$$

Comme les $b_{im}^j(p)$ sont des probabilités discrètes, un histogramme est calculé pour chaque composante pendant la phase d'apprentissage, et leur estimation est donnée par le nombre de vecteurs dans l'état s_i^j ayant l'observation p dans la $m^{ème}$ composante divisé par le nombre de vecteurs dans l'état s_i^j.

Après l'exécution de l'algorithme de Viterbi et la détermination du chemin optimal, un module de post-traitement déduit les durées pour chaque super-état. La mesure de distorsion bayésienne P^* est pondérée par les probabilités de ces durées pour les super-états. A chaque super-état est associée une loi de distribution de probabilités de durée gaussienne dont les paramètres sont estimés pendant l'apprentissage.

Gilloux [Gil 94] a proposé pour les chiffres manuscrits une autre topologie de PHMMs où les super-états sont des classes d'équivalence de super-états (voir figure 7). En effet, la topologie classique telle qu'elle a été utilisée par Agazzi pour l'imprimé fait l'hypothèse que les lignes consécutives sont indépendantes. Gilloux pense que l'utilisation de plusieurs super-états par classe et donc des probabilités de transition différentes entre éléments de la même classe doit permettre de représenter la corrélation entre probabilités d'émission de lignes voisines. Ce modèle reste cependant complexe à mettre en œuvre surtout pour établir les transitions entre super-états à l'intérieur d'une même classe et entre les classes. Il a été testé sur des chiffres manuscrits, recueillis sur des codes postaux. Le modèle comporte 5 classes de 10 super-états et 6 états par ligne. 4891 images de chiffres normalisés à 16×16 pixels ont servi pour l'apprentissage. Le test a été effectué sur un autre ensemble de 4891 images et conduit à un taux jugé moyen par l'auteur de 90%.

Dans le cadre d'une étude sur la reconnaissance de mots arabes imprimés, nous avons proposé une architecture de PHMM fondée sur une exploitation de la durée d'observation aussi bien dans les modèles secondaires que dans le modèle principal [Ben 96].

L'architecture des modèles est spécifique à la morphologie des PAW (Piece of Arabic Word) qui composent les mots arabes.

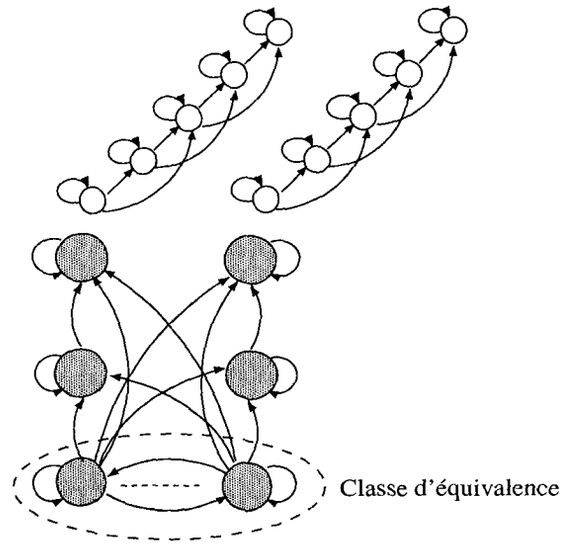


Figure 7. – Modèle de PHMM, d'après Gilloux.

Ainsi, l'image d'un PAW se décompose verticalement en cinq zones correspondant respectivement de haut en bas à l'emplacement de la hampe, des signes diacritiques supérieurs, de la bande centrale, des signes diacritiques inférieurs et du jambage. A chaque zone est associé un super-état auquel correspond un HMM horizontal modélisant la zone concernée.

Les observations dans les HMMs horizontaux sont constituées chacune de la couleur du segment courant (noire ou blanche), de sa longueur et de son emplacement relatif par rapport au segment qui se trouve au-dessus. Ainsi, la durée dans les états secondaires représente l'observation de la longueur des segments.

Le HMM vertical est un modèle à durée explicite pour mieux prendre en compte la hauteur des cinq bandes indiquées. Ainsi, la durée dans un super-état est assimilée au nombre de lignes analysées par lui.

Lors de l'apprentissage, l'image est segmentée en bandes donnant ainsi les durées et les lignes aférentes aux super-états. La distribution de probabilités de la durée est calculée par estimation de la fréquence d'une certaine hauteur de bande pour un super-état donné. La probabilité de transition entre deux super-états successifs $j-1$ et j est estimée comme suit :

$$a_{j-1j} = \frac{\sum_{k=1}^{K} \frac{d_j^k}{d_{j-1}^k + d_j^k}}{K}, \qquad j = 2 \ldots N \qquad (39)$$

où K est le nombre d'échantillons considérés pour l'apprentissage d'un PAW donné, d_{j-1}^k et d_j^k sont respectivement, la durée associée au super-état $j-1$ et j pour l'échantillon k.

Lors de la reconnaissance, le système tente de retrouver les bandes maximisant à la fois la probabilité d'émission des lignes de la bande et la probabilité de sa hauteur (durée). La formule de

récursion de l'algorithme de Viterbi est :

$$\Delta_y(j) = \max\{\Delta_{y-1}(j-1)a_{j-1j}, \Delta_{y-1}(j)P(d_j+1|s_j)\}P_j(y), \quad 1 \leq j \leq N,\ y = 2 \ldots Y \quad (40)$$

où :

$$d_j = \begin{cases} d_j + 1, & \text{si on reste dans } s_j \\ 0, & \text{si on effectue la transition } s_{j-1} \rightarrow s_j \end{cases}$$

Le système a été testé sur une base restreinte de PAWs (12 types de PAWs, deux fontes distinctes et différents corps, 100 à 130 échantillons par PAW). Les résultats sont prometteurs (99.70%) mais restent à valider sur une base beaucoup plus conséquente. Cela étant, le choix de reconnaissance par PAW limitera l'utilisation du système à des applications à vocabulaire réduit. Toutefois, l'originalité de ce système par rapport aux précédents, outre sa simplicité de mise en œuvre et sa faible complexité (nombre d'états réduit), se place à deux niveaux :

– Le codage des observations par des segments le long des lignes traduit d'une part une notion de durée horizontale et tente à régler d'autre part, à l'aide de l'information de décalage de segments voisins, la corrélation entre lignes successives.

– La prise en compte explicitement d'une durée verticale estimant la hauteur des bandes.

3.3. décomposition probabiliste

On se place dans la situation décrite dans 2.3.2. Dans ce cas, la forme F est décomposable en sous-formes selon deux directions et il existe un modèle η_E associé à l'étiquette E. La probabilité *a posteriori* de l'étiquette E par rapport à la forme F est :

$$P(E|F) = P(\eta_E|F) \propto P(F|\eta_E)P(\eta_E) \quad (41)$$

où $P(\eta_E)$ est la probabilité *a priori* du modèle η_E. Son estimation se fait comme dans 2.3.2.

Le terme $P(F|\eta_E)$ se décompose en une somme sur toutes les séquences de super-états du produit de la probabilité conditionnelle de la forme sachant la séquence de super-états et de la probabilité de la séquence sachant le modèle.

$$P(F|\eta_E) = P(f_1 \ldots f_Y|\eta_E) = \sum_{q_1,\ldots,q_Y} P(f_1,\ldots,f_Y|q_1,\ldots,q_Y,\eta_E)P(q_1,\ldots,q_Y|\eta_E) \quad (42)$$

D'une manière analogue que pour le cas 1D, on suppose qu'un seul chemin $\{q_y^*\}$ contribue au calcul de $P(F|\eta_E)$, ce qui donne :

$$P(F|\eta_E) = P(f_1 \ldots f_Y|\eta_E) = = P(f_1,\ldots,f_Y|q_1^*,\ldots,q_Y^*,\eta_E)P(q_1^*,\ldots,q_Y^*|\eta_E) \quad (43)$$

$\{q_y^*\}$ est toujours un processus stochastique stationnaire d'ordre 1 qui vérifie l'équation (16).

En faisant l'hypothèse d'indépendance des lignes, $P(f_1,\ldots,f_Y|q_1^*,\ldots,q_Y^*,\eta_E)$ se développe dans le cas des PHMMs comme suit :

$$P(f_1,\ldots,f_Y|q_1^*,\ldots,q_Y^*,\eta_E) = = \prod_{y=1}^{Y} P(f_y|q_y^*,\eta_E) = \prod_{y=1}^{Y} P(f_y|\lambda_E^{q_y^*}) \quad (44)$$

La quantité $P(f_y|\lambda_E^{q_y^*})$ représente la probabilité d'émission de la ligne y de la forme F par le sous-modèle $\lambda_E^{q_y^*}$ du PHMM η_E. Elle peut être développée comme montré dans le cas 1D par les équations (14), (15) et (16).

4. champs de Markov causaux

Les PHMMs associent plusieurs lignes à chaque super-état, faisant l'hypothèse que plusieurs lignes consécutives sont étroitement corrélées entre elles et donc, peuvent être analysées par un même HMM [Aga 93]. Cette hypothèse d'indépendance facilite la mise en œuvre mais ne garantit pas l'optimalité du modèle pendant la reconnaissance car elle n'est pas toujours vérifiée.

Nous pensons qu'une modélisation parfaitement bidimensionnelle de l'image serait plus profitable. Aussi, nous avons étudié l'applicabilité des champs aléatoires de Markov à la reconnaissance de chiffres et de mots manuscrits non-contraints [Sao 96b, Sao 96a]. Contrairement aux PHMMs, les champs aléatoires possèdent une véritable structure 2D [Gem 84] dans la mesure où la probabilité d'un site du champ est conditionnée par les sites voisins et conditionne à son tour celles d'autres sites. Nous avons limité notre étude aux champs causaux sur lesquels il est possible d'induire une relation d'ordre de type lexicographique en restreignant les voisinages à ceux permettant une progression naturelle du calcul de la probabilité de masse cumulée du champ (joint field mass probability). Deux types de chaînes de Markov causales sont largement utilisées en traitement d'images : les réseaux de Markov [Abe 65] (Markov Random Mesh ou MRM) et les champs de Markov unilatéraux [Pre 75] (Non-Symmetric Half-Plane Markov chains ou NSHP). Ces deux modèles causaux diffèrent par le choix des états locaux et de leur « passé » (le voisinage qui conditionne la probabilité d'une variable aléatoire). Ils sont équivalents si le passé est réduit au quart supérieur gauche du plan, mais le NHSP peut accepter un passé beaucoup plus étendu, ce qui est utile dans plusieurs applications de traitement d'images [Jen 87].

Après un bref rappel de la définition des NHSPs, nous montrerons comment la décomposition probabiliste peut se faire dans ce contexte. Une application de ces modèles à la reconnaissance de chiffres et de mots manuscrits que nous avons réalisée illustrera leur intérêt en RAE.

4.1. champs de Markov unilatéraux

Soit L un treillis rectangulaire de $m \times n$ sites. On définit sur L un champ aléatoire associé à la forme $F = \{f_{ij}\}_{1 \leq i \leq m,\ 1 \leq j \leq n}$ où les f_{ij} sont des variables aléatoires. On notera par f_j les colonnes du champ F. On fait correspondre à chaque site $(i,j) \in L$ les ensembles suivants :

$$\Sigma_{ij} = \{(k,l) \in L \mid l < j \text{ ou } (l = j,\ k < i)\},\ \ \Theta_{ij} \subset \Sigma_{ij} \quad (45)$$

Σ_{ij} est appelé demi-plan non symétrique et Θ_{ij} le support du site (i,j). La figure 8 illustre ces deux types d'ensembles. La notation $P(f_{ij}|f_A)$, $A \subset L$ signifie $P(f_{ij}|f_{kl})$, $(k,l) \in A$.

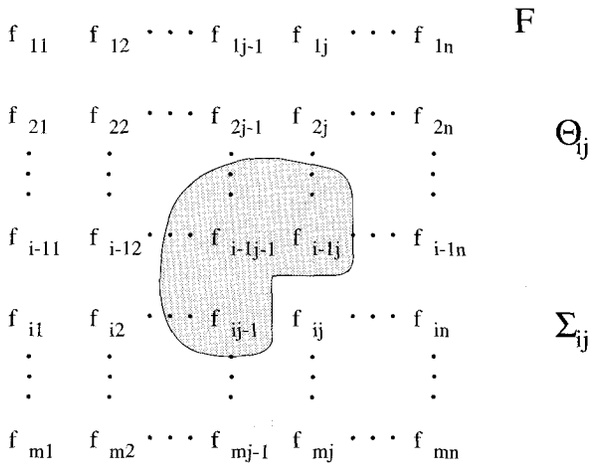


Figure 8. – Ensembles de sites associés à un point (i,j).

Définition 8.

F est un champ aléatoire unilatéral de Markov si et seulement si :

$$P(f_{ij}|f_{\Sigma_{ij}}) = P(f_{ij}|f_{\Theta_{ij}}), \qquad \forall\,(i,j) \in L \quad (46)$$

Le calcul de la probabilité de masse cumulée pour tous les sites du champ aléatoire F a la propriété suivante :

$$P(F) = \prod_{(i,j)\in L} P(f_{ij}|f_{\Sigma_{ij}}) = \prod_{(i,j)\in L} P(f_{ij}|f_{\Theta_{ij}}) \quad (47)$$

Une manière directe de calculer $P(F)$ est donc :

$$P(F) = \prod_{j=1}^{n}\prod_{i=1}^{m} P(f_{ij}|f_{\Theta_{ij}}) \quad (48)$$

Ceci soulève une question importante relative à la détermination des supports Θ_{ij}. Une solution largement répandue consiste à fixer la même forme pour ces ensembles pour tous les sites (i,j) avec des conditions aux frontières adéquates (comme c'est, par exemple, le cas des champs de Pickard [Der 89]).

4.2. décomposition probabiliste et introduction du modèle

Dans ce cadre, nous gardons la notion de probabilité conditionnelle d'une forme F sachant une étiquette E. La seule décomposition étudiée est celle par rapport à un modèle ψ_E associé à l'étiquette E. L'équation (48) devient :

$$P(F|E) = P(F|\psi_E) = \prod_{j=1}^{n}\prod_{i=1}^{m} P(f_{ij}|\psi_E, f_{\Theta_{ij}}) \quad (49)$$

En considérant la réalisation du champ aléatoire (image binaire de la forme) comme une séquence d'observations de colonnes, la chaîne de Markov NSHP peut être implémentée par un HMM séquentiel gauche-droite. Dans un état du HMM, la probabilité d'observation est donnée par le produit le long de la colonne des probabilités conditionnelles de pixels. Une transition d'un état du modèle à un autre aura pour effet le changement de l'ensemble des distributions et, par conséquent, de l'adaptabilité du modèle aux différentes primitives dans l'image. Ces primitives seront associées aux états du modèle, suite à la phase d'apprentissage. La figure 9 illustre le schéma d'implémentation d'un tel modèle.

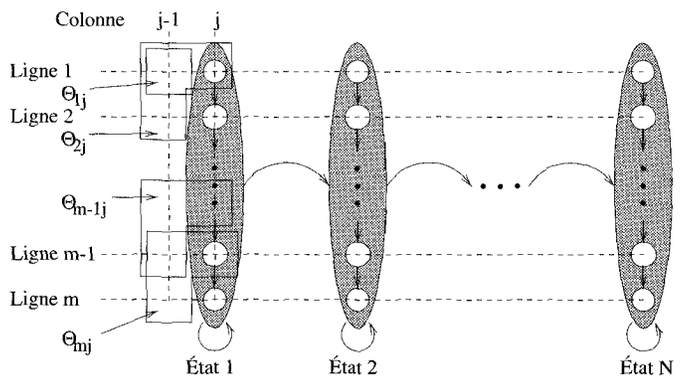


Figure 9. – Architecture d'un modèle NSHP-HMM d'ordre 3.

Soit $Q^* = q_1^* \ldots q_n^*$ le chemin d'états qui contribue le plus au calcul de la vraisemblance de F. En introduisant la convention

$P(q_1^*|q_0^*) = P(q_1^*)$ (pour des raisons de lisibilité), l'équation (49) devient :

$$\begin{aligned} P(F|\psi_E) &= P(f_1,\ldots,f_n|q_1^*,\ldots,q_n^*,\psi_E)P(q_1^*,\ldots,q_n^*|\psi_E) \\ &= \prod_{j=1}^{n} P(f_j|f_1,\ldots,f_{j-1},q_j^*,\psi_E)P(q_j^*|q_{j-1}^*) \\ &= \prod_{j=1}^{n} P(q_j^*|q_{j-1}^*)\prod_{i=1}^{m} P(f_{ij}|f_{\Sigma_{ij}},q_j^*,\psi_E) \\ &= \prod_{j=1}^{n} P(q_j^*|q_{j-1}^*)\prod_{i=1}^{m} P(f_{ij}|f_{\Theta_{ij}},q_j^*,\psi_E,) \end{aligned}$$

L'interprétation de l'équation (50) nécessite de préciser les points suivants :

– Le processus stochastique qui modélise le chemin d'états est supposé être markovien du premier ordre :

$P(q_1^*,\ldots,q_n^*|\psi_E) = \prod_{j=1}^{n} P(q_j^*|q_{j-1}^*)$.

– Pour la colonne f_j, les distributions de probabilités conditionnelles dépendent uniquement de l'état q_j^*.

Afin de définir complètement le modèle tout en restant conforme aux notations employées pour un HMM ordinaire (discret), nous donnons les éléments suivants :

• $\Theta = \{\Theta_{ij}\}_{(i,j)\in L}$, $\Theta_{ij} = \{(i-i_k, j-j_k)|1 \le k \le P, j_k > 0 \text{ ou } (j_k = 0, i_k > 0)\} \cap L$, où P est le nombre de voisins par pixel. Θ représente l'ensemble des voisinages et P l'ordre du modèle. Les Θ_{ij} sont supposés être ordonnés.

• $B = \{b_{ik}(f,\mathbf{f})\}_{1\le i\le m, 1\le k\le N}$, $f \in \{0,1\}$, $\mathbf{f} \in \{0,1\}^P$, $b_{ik}(f,\mathbf{f}) = P(f_{ij} = f|f_{\Theta_{ij}} = \mathbf{f}, q_j = s_k)$, les probabilités conditionnelles d'observation de pixels.

Le modèle NSHP-HMM ainsi défini est représenté par le quadruplet $\lambda = (\Theta, A, B, \pi)$ où A représente la matrice de probabilités de transition entre états et π les probabilités initiales.

Dans la suite, nous allons montrer comment s'effectue l'estimation de la probabilité d'émission d'une image (sa plausibilité) et donner quelques éléments relatifs à l'apprentissage et à la reconnaissance.

4.3. apprentissage et reconnaissance

Une évaluation optimale de la vraisemblance $P(F|\lambda)$ est obtenue en utilisant des variantes des fonctions *forward-backward*. Nous définissons la fonction *forward* α (la fonction β suivant une définition duale) comme étant la probabilité cumulée du champ jusqu'à la colonne j à l'état s_i, $\alpha_j(i) = P(f_1, f_2, \ldots, f_j, q_j = s_i|\lambda)$:

$$\begin{aligned} \alpha_1(i) &= \pi_i \prod_{k=1}^{m} b_{ki}(f_{k1}, f_{\Theta_{k1}}), \qquad 1 \le i \le N \\ \alpha_j(i) &= \left[\sum_{l=1}^{N} \alpha_{j-1}(l)a_{li}\right] \prod_{k=1}^{m} b_{ki}(f_{kj}, f_{\Theta_{kj}}),\ j = 2\ldots n \\ P(F|\lambda) &= \sum_{i=1}^{N} \alpha_n(i) \end{aligned} \tag{51}$$

Pendant l'apprentissage, on détermine les paramètres (A, B, π) du modèle qui maximisent la quantité $\prod_{k=1}^{K} P(F^k|\lambda)$, où F^k sont les échantillons utilisés. Nous pouvons noter que, comme dans le cas unidimensionnel, il n'y a pas de critère global d'optimisation et de méthode directe. Nous utilisons alors le même critère de maximisation de la vraisemblance (MLE), en appliquant la ré-estimation de Baum-Welch. Nous allons uniquement détailler la ré-estimation des probabilités conditionnelles d'observation de pixels :

$$\bar{b}_{il}(f,\mathbf{f}) = \begin{cases} \dfrac{\displaystyle\sum_{k=1}^{K} \frac{1}{P_k} \sum_{\substack{j=1 t.q. \\ f_{ij}^k = f \text{ et} \\ f_{\Theta_{ij}}^k = \mathbf{f}}}^{n_k} \alpha_j^k(l)\beta_j^k(l)}{\displaystyle\sum_{k=1}^{K} \frac{1}{P_k} \sum_{\substack{j=1\ t.q. \\ f_{\Theta_{ij}}^k = \mathbf{f}}}^{n_k} \alpha_j^k(l)\beta_j^k(l)} & \text{dénominateur} \neq 0 \\ b_{il}(f,\mathbf{f}), & \text{sinon} \end{cases} \tag{52}$$

$$f \in \{0,1\}, \qquad \mathbf{f} \in \{0,1\}^P, \qquad 1 \le i \le m, \qquad 1 \le l \le N$$

où $P_k = P(F^k|\lambda)$ représente la probabilité d'émission de l'échantillon F^k et n_k sa longueur.

Si l'on examine l'équation (52), nous pouvons remarquer que la ré-estimation se fait par un comptage MLE du nombre de fois qu'une certaine configuration de pixels est rencontrée et ceci pour toutes les configurations, toutes les lignes et tous les états du modèle. Notons enfin que tous les échantillons sont supposés avoir le même nombre de lignes m, ce qui nécessite évidemment une normalisation en hauteur des images avant les étapes d'apprentissage ou de reconnaissance.

En ce qui concerne la reconnaissance, nous avons opté pour une approche de type *modèle discriminant* en construisant un modèle NSHP-HMM pour chaque classe. Le résultat est obtenu en calculant la vraisemblance de la forme entrante pour chaque modèle et en étiquetant l'image par le modèle qui produit la probabilité a posteriori maximale.

4.4. expérimentations et résultats

Une première expérimentation a été conduite sur une base multi-scripteurs de 562 chiffres manuscrits. 337 d'entre eux choisis de manière aléatoire ont servi à l'apprentissage des modèles et le reste (225), pour le test. Toutes les images ont été ramenées à une hauteur de $m = 16$ lignes. Les modèles utilisés sont d'ordre $P = 2$ avec le même nombre d'états $N = 10$ (voir figure 10). Nous avons obtenu un taux de reconnaissance (première réponse) de 98.22% et un taux maximal pour les trois premières réponses.

Une seconde expérimentation a été opérée sur une base de mots réels du SRTP[1], extraits de montants littéraux de chèques postaux (7057 images de mots écrits par 1779 scripteurs et un vocabulaire de 27 mots). Il est évident que la tâche du système est beaucoup plus difficile que dans le cas des chiffres car, s'agissant d'écritures non contraintes, les mots présentent des distorsions plus importantes. Cependant, le comportement du système reste inchangé voyant les mots comme des formes binaires quelconques. Cette adaptation du système à des formes différentes (chiffres, mots, etc.) constitue le point fort du type de modélisation adopté.

5284 images (approximativement 3/4 de la base) ont été choisies aléatoirement pour l'apprentissage des modèles. La reconnaissance a été opérée sur 1773 images de mots et atteint le score de 89.68% en première réponse sans la prise en compte de la syntaxe des montants. Les paramètres initiaux qui ont été choisis pour chaque modèle de mot sont :

– *Nombre d'états* : il est proportionnel au nombre moyen de colonnes des images, $\overline{n}$, après normalisation en hauteur. En pratique, un nombre d'états égal à $\overline{n}/2$ (variant de 11 à 35 pour $m = 20$ lignes) donne les meilleurs résultats.

– *Transitions entre états* : nous autorisons uniquement les transitions vers le même état et entre états successifs. Initialement, les transitions sont équiprobables : $a_{ii} = a_{ii+1} = 0.5,\ 1 \leq i \leq N-1$.

– *Nombre de lignes* : pour des raisons de complexité de calcul, des expérimentations ont été faites pour $m = 10$, $m = 15$ et $m = 20$.

– *Ordre du modèle (nombre de pixels voisins)* : des expérimentations ont été effectuées pour $P = 0 \ldots 4$ correspondant aux configurations de voisinages illustrées dans la figure 10.

– *Probabilités conditionnelles d'observation de pixels* : toutes les images d'un même mot ont été divisées en N bandes verticales de largeur égale. Un comptage normalisé du nombre de configurations de pixels $f_{ij}^k = f$ et $f_{\Theta_{ij}}^k = \mathbf{f}, \forall f \in \{0,1\}, \mathbf{f} \in \{0,1\}^P$, est effectué à l'intérieur de chaque bande pour tous les échantillons F^k.

La figure 11 donne un apperçu visuel des capacités réelles d'apprentissage des modèles. Les niveaux de gris codent la probabilité d'apparition des pixels noirs relevée dans tous les états du modèle et cela pour toutes les lignes. Les prototypes de mots ont été obtenus à l'aide de modèles d'ordre 3 entraînés avec des échantillons de hauteur $m = 30$, au bout de 20 itérations. Cette figure montre clairement que, malgré l'énorme variabilité des formes au niveau du pixel, les modèles sont capables de faire ressortir les agglomérations de pixels caractéristiques de l'écriture.

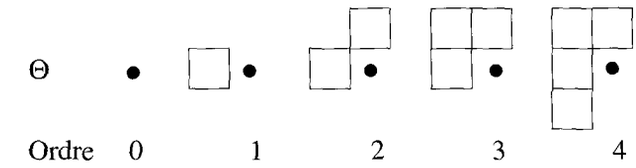


Figure 10. – Différents voisinages considérés.

Figure 11. – Synthèse de prototypes de chiffres et de mots.

5. conclusion

Dans ce papier, nous avons tenté d'unifier les mécanismes de la reconnaissance probabiliste utilisés dans les modèles de Markov. Pour cela, les termes apparaissant dans la formule de Bayes sont décomposés de manières différentes suivant la dimensionnalité de la forme et les hypothèses de dépendance entre sous-formes et étiquettes. On distingue deux grands cas de décomposition :

[1] Service de Recherche Technique de la Poste

de la forme par rapport à l'étiquette et de la forme par rapport à un modèle associé à l'étiquette. Le premier cas est utilisé typiquement dans les applications de RAE de niveaux supérieurs (lexical et syntaxique), étant plus spécifique aux HMMs 1D. Le deuxième cas connaît plusieurs développements en fonction de la dimensionnalité du modèle et de l'interprétation de la vraisemblance de la forme. Dans ce cas, on associe à la forme un processus stochastique représentant le chemin d'états du modèle qui permet d'observer le mieux les sous-formes. La probabilité d'une sous-forme peut être conditionnée dans le cas 1D soit par l'état courant (HMM discret ou continu), soit par deux états successifs (courant et précédent), soit enfin par l'état courant et la sous-forme précédente. Dans le cas 2D, la probabilité de la sous-forme est conditionnée par la sous-forme précédente selon un axe d'analyse ; les résultats de l'analyse sur cet axe étant corrélés globalement par un autre HMM (le cas des PHMMs), et pour les champs aléatoires causaux, par un voisinage bidimensionnel de sous-formes.

Nous avons étudié dans cet article quelques types de décompositions favorisant l'utilisation de modélisations markoviennes d'ordre et de dimension variables. Pour des formes réelles comme l'écriture, ces décompositions traduisent des connaissances contextuelles qu'il faut expliciter avant toute modélisation. Le défi permanent dans ce domaine reste orienté vers la recherche de contextes plus riches et de modèles de représentation stochastique appropriés de ces contextes.

BIBLIOGRAPHIE

[Abe 65] K. Abend, T. J. Harley, and L. N. Kanal. Classification of Binary Random Patterns. *IEEE Trans. Inform. Theory*, 1(11) : 538–544, 1965.

[Aga 93] O. E. Agazzi and S. Kuo. Hidden Markov Model Based Optical Character Recognition in the Presence of Deterministic Transformation. *Pattern Recognition*, 26(12) : 1813–1826, February 1993.

[Ani 91] J. C. Anigbogu and A. Belaïd. Recognition of Multifont Text Using Markov Models. In 7^{th} *Scandinavian Conference on Image Analysis*, volume I, pages 469–476, August 1991.

[Ani 95] J. C. Anigbogu and A. Belaïd. Hidden Markov Models in Text Recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 9(5), 1995.

[Bah 83] L. R. Bahl, F. Jelinek, and R. L. Mercer. A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on PAMI*, 5(2) : 179–190, March 1983.

[Bah 86] L. R. Bahl, P. F. Brown, P. V. de Souza, and R. L. Mercer. Maximum Mutual Information Estimation of Hidden Markov Model Parameters for Speech Recognition. In *IEEE-ICASSP*, volume 1, pages 49–52. IEEE, 1986.

[Bah 88] L. R. Bahl, P. F. Brown, P. V. de Souza, and R. L. Mercer. A New Algorithm for the Estimation of Hidden Markov Model Parameters. In *IEEE-ICASSP*, volume 1, pages 493–496, 1988.

[Bau 67] L. E. Baum and J. A. Egon. An Inequality with Applications to Statistical Estimation for Probabilistic Functions of a Markov Process and to a Model for Ecology. Bull, Ams. volume 73, pages 360–363, 1967.

[Bau 68] L. E. Baum and T. Petrie. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *Annals of Mathematics and Statistics*, 37 : 1554–1563, 1968.

[Bau 72] L. E. Baum. *An Inequality and Associated Maximization Technique for Probabilistic Functions of Markov Process*. Inequalities, 3, pages 1–8, 1972.

[Bel 94a] A. Belaïd and J. C. Anigbogu. Mise à contribution de plusieurs classifieurs pour la reconnaissance de textes multifonte. *Traitement du signal*, 11(1) : 57–75, 1994.

[Bel 94b] A. Belaïd and G. Saon. Use of stochastic models in text recognition. In *KOSEF-CNRS French-South Korean Workshop on Text Recognition*, pages 79–98, September 1994.

[Ben 96] N. BenAmara and A. Belaïd. Printed PAW Recognition Based on Planar Hidden Markov Models. In 13^{th} *International Conference on Pattern Recognition*, Viena, Austria, 1996.

[Ber 92] S. Bercu, B. Delyon, and G. Lorette. Segmentation par une méthode de reconnaissance d'écriture cursive en-ligne. In A. Belaïd, editor, *Traitement de l'écriture et des documents, Actes du colloque CNED'92*, pages 144–151, Nancy, juillet 1992.

[Ber 93] S. Bercu and G. Lorette. On-line Handwritten Word Recognition : an Approach Based on Hidden Markov Models. In *Proc. of the 3rd International Workshop on Frontiers in Handwriting Recognition*, pages 385–390, 1993.

[Bos 94] C. B. Bose and S. Kuo. Connected and Degraded Text Recognition Using Hidden Markov Model. *Pattern Recognition*, 27(10) : 1345–1363, 1994.

[Boz 89] R. M. Bozinovic and S. N. Srihari. Off-Line Cursif Script Word Recognition. *IEEE Transactions on PAMI*, 11, Jan. 1989.

[Che 92] M. Y. Chen, A. Kundu, J. Zhou, and S. N. Srihari. Off-Line Handwritten Word Recognition Using Hidden Markov Model. In *USPS'92*, 1992.

[Che 93a] M. Y. Chen and A. Kundu. An Alternative to Variable Duration HMM in Handwritten Word Recognition. In *Proc. IWFHR-3*, pages 82–91, Paris, 1993.

[Che 93b] M. Y. Chen, A. Kundu, and S. N. Srihari. Handwritten Word Recognition Using Continous Density Variable Duration Hidden Markov Model. In *Second International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 105–108, Tsukuba, Japan, 1993.

[Coh 94] E. Cohen, J. J. Hull, and S. N. Srihari. Control Structure for Interpreting Handwritten Addresses. *IEEE Transactions on PAMI*, 16(5) : 1049–1055, 1994.

[Cre 94] J.P. Crettez. Premier degré de caractérisation des écritures manuscrites : essai de regroupement des écritures en familles. In *Actes du* $3^{ème}$ *Colloque National sur l'Écrit et le Document*, pages 71–81, Rouen, juillet 1994.

[Cre 96] J. P. Crettez, M. Gilloux, and M. Leroux. Que dit la "Main" du scripteur aux "Yeux" du lecteur. In *Actes du* $4^{ème}$ *Colloque National sur l'Écrit et le Document*, pages 291–296, Nantes, France, July 1996.

[Der 89] H. Derin and P. A. Kelly. Discrete-Index Markov-Type Random Processes. *Proceedings of the IEEE*, 77(10) : 1485–1510, 1989.

[Eph 87] Y. Ephraim, A. Dembo, and L. R. Rabiner. A Minimum Discrimination Information Approach for Hidden Markov Modeling. In *IEEE-ICASSP*, volume 1, pages 25–28. IEEE, 1987.

[For 73] G. D. Forney. The Viterbi Algorithm. *Proceedings of the IEEE*, 661(3), 1973.

[Gem 84] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Transactions on PAMI*, 6(6) : 721–741, 1984.

[Gil 92a] A. M. Gillies. Cursive Word Recognition Using Hidden Markov Models. In *USPS'92*, pages 557–562, 1992.

[Gil 92b] M. Gilloux and M. Leroux. Approches markoviennes en reconnaissance de l'écriture cursive. In A. Belaïd, editor, *Traitement de l'écriture et des documents, Actes du colloque CNED'92*, pages 152–159, Nancy, juillet 1992.

[Gil 92c] M. Gilloux and M. Leroux. Recognition of Cursive Script Amounts on Postal Cheques. In *USPS advanced Technology Conference*, pages 545–556, 1992.

[Gil 93] M. Gilloux, M. Leroux, and J. M. Bertille. Strategies for Handwritten Words Recognition Using Hidden Markov Models. In *Second International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 299–304, Tsukuba, City Science Japan, 1993.

[Gil 94] M. Gilloux. Reconnaissance de chiffres manuscrits par modèle de Markov pseudo-2D. In *Actes du 3ème Colloque National sur l'Écrit et le Document*, pages 11–17, Rouen, France, 1994.

[Gop 89] P.S. Gopalakrishnan *and al.* A Generalization of the Baum Algorithm to Rational Objective Functions. In *IEEE-ICASSP*, volume 2, pages 631–634. IEEE, 1989.

[Hul 83] J. J. Hull, S. N. Srihari, and R. Choudhari. An Integrated Algorithm for Text Recognition : Comparison with a Cascaded Algorithm. *IEEE Transactions on PAMI*, PAMI-5(4) : 384–395, July 1983.

[Jel 80] F. Jelinek and R. L. Mercer. Interpolated Estimation of Markov Source Parameters from Sparse Data. *Workshop on Pattern Recognition in Practice*, March 1980.

[Jen 87] F. C. Jeng and J. W. Woods. On the Relationship of the Markov Mesh to the NSHP Markov Chain. *Pattern Recognition Letters*, 5(4) : 273–279, 1987.

[Kri 90] A. Kriouile, J. F. Mari, and J. P. Haton. Some Improvements in Speech Recognition Algorithms Based on HMM. In *IEEE-ICASSP*, pages 545–548. IEEE, 1990.

[Kuo 94] S. Kuo and O. E. Agazzi. Keyword Spotting in Poorly Printed Documents Using Pseudo 2-D Hidden Markov Models. *IEEE Transactions on PAMI*, 16(8) : 842–848, 1994.

[Lem 93] B. Lemarié. Practical Implementation of A Radial Basis Function Network For Handwritten Digit Recognition. In *Second International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 412–415, Tsukuba, Japan, 1993.

[Lem 96] B. Lemarié, M. Gilloux, and M. Leroux. Un modèle neuro-markovien contextuel pour la reconnaissance de l'écriture manuscrite. In *Actes 10ème Congrès AFCET de Reconnaissance des Formes et Intelligence Artificielle*, Rennes, France, 1996.

[Lev 83] S. E. Levinson, L. R. Rabiner, and M. M. Shondi. An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition. *The Bell System Technical Journal*, 62(4), 1983.

[Lev 92] E. Levin and R. Pieraccini. Dynamic Planar Warping for Optical Character Recognition. In *IEEE-ICASSP*, volume III, pages 149–152. IEEE, 1992.

[Mer 88] B. Merialdo. Phonetic Recognition Using Hidden Markov Models and Maximum Mutual Information Training. In *IEEE-ICASSP*, volume 9, pages 111–114. IEEE, 1988.

[Neu 75] D. L. Neuhoff. The Viterbi Algorithm as an Aid in Text Recognition. *IEEE Transactions on Information Theory*, pages 222–226, March 1975.

[Nor 94] Y. Normandin, R. Cardinand, and R. De Mori. High-Performance Connected Digit Recognition Using Maximum Mutual Information Estimation . *IEEE Transactions on Speech and Audio Processing*, 2(2) : 299–311, April 1994.

[Pre 75] D. Preuss. Two-Dimensional Facsimile Source Coding Based on a Markov Model. *NTZ 28*, 5(4) : 358–363, 1975.

[Rab 89] L. R. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), February 1989.

[Sao 94] G. Saon, A. Belaïd, and Y. Gong. Off-line handwriting recognition by statistical correlation. In *MVA'94 IAPR Workshop on Machine Vision Applications*, pages 371–374, Japan, 1994.

[Sao 95] G. Saon, A. Belaïd, and Y. Gong. Stochastic Trajectory Modeling for Recognition of Unconstrained Handwritten Words. In *Third International Conference on Document Analysis and Recognition (ICDAR'95)*, pages 508–511, Montréal, Canada, 1995.

[Sao 96a] G. Saon and A. Belaïd. High Performance Unconstrained Word Recognition System Combining HMMs and Markov Random Fields. *International Journal of Pattern Recognition and Artificial Intelligence Special Issue on Automatic Bankcheck Processing*, 1996. Accepted for publication.

[Sao 96b] G. Saon and A. Belaïd. Recognition of Unconstrained Handwritten Words Using Markov Random Fields and HMMs. In *Fifth International Workshop on Frontiers in Handwriting Recognition (IWFHR5)*, University of Essex, England, September 1996. In press.

[Shi 79] R. Shinghal and G. T. Toussaint. Experiments in Text Recognition with Modified Viterbi Algorithm. *IEEE Transactions on PAMI*, 2(1) : 184–192, March 1979.

[Sri 84] S. N. Srihari. *Computer Text Recognition and Error Correction*. IEEE Computer Society Press, Silver Spring, MD, 1984.

[Sri 93] S. N. Srihari, V. Govindaraju, and A. Shekhawat. Interpretation of Handwritten Addresses in US Mailstream. In *Second International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 291–294, Tsukuba, Japon, Oct. 20 – 22 1993.

[Tou 78] G. T. Toussaint. The Use of Context in Pattern Recognition. *Pattern Recognition*, 10 : 189–204, 1978.

Manuscrit reçu le 19 juin 1996

LES AUTEURS

Abdel BELAID

A. Belaïd a fait ses études supérieures à l'Université Louis Pasteur (ULP) à Strasbourg de 1972 à 1976, puis a rejoint l'Université Henri Poincaré (Nancy 1) et le CRIN (Centre de Recherche en Informatique de Nancy) où il a obtenu sa thèse de 3e cycle en 1979 et sa thèse d'état en 1987. Après quelques années d'enseignement en tant qu'assistant, puis maître assistant à l'Université UHP, il est Chargé de recherche au CNRS depuis 1984. Ses domaines de recherche sont le traitement d'images et la reconnaissance des formes où il a publié une cinquantaine d'articles. Il est co-auteur d'un livre intitulé : Reconnaissance des formes : méthodes et applications. Il anime un groupe de recherche au CRIN autour de plusieurs projets sur l'analyse de documents et la reconnaissance de l'écriture et il est auteur de plus de 50 articles. Il est membre du Groupement de Recherche en Communication Ecrite (GRCE), de SPECIF et de l'Association Française pour la Cybernétique Economique et Technique (AFCET).

George SAON

G. Saon a étudié à la Faculté d'Automatique et Ordinateurs (Institut Polytechnique de Bucarest) où il a obtenu le diplôme d'Ingénieur en Informatique, section Soft en 1995. En 1993, il a également obtenu une licence en Mathématiques Pures délivrée par l'Université de Bucarest. Ensuite, il a fait son DEA à l'Université Henri Poincaré Nancy 1 et actuellement il est en troisième année de thèse de doctorat sur l'utilisation des modèles stochastiques dans la reconnaissance de l'écriture manuscrite. Il est membre du Centre de Modélisation et Calcul à Hautes Performances Charles Hermite.