Apprentissages discriminants en reconnaissance de mots cursifs en-ligne

E. CAILLAULT[1], C. VIARD-GAUDIN[1]

[1] IRCCyN – UMR CNRS 6597, Image et Video Communications

Ecole polytechnique de l'université de Nantes, rue Christian Pauc, BP 50609, FR-44306 NANTES cedex 3

{emilie.caillault,christian.viard-gaudin}@univ-nantes.fr

Résumé – Ce papier décrit différents modes d'apprentissage de systèmes hybrides basés sur un schéma neuro-markovien (TDNN multi-états – HMM) appliqués à la reconnaissance de mots cursifs saisis en-ligne. Nous avons considéré différentes fonctions de coût, incluant à la fois des critères d'information mutuelle (MMI) avec un apprentissage discriminant et une estimation du maximum de vraisemblance, pour entraîner le système globalement au niveau mot. Nous avons analysé l'impact de la modélisation markovienne en variant de un à trois le nombre d'états d'un modèle de Markov caché lettre HMM lettre). Plusieurs expérimentations sur ces critères et modélisations ont été menées sur la base IRONOFF dans un contexte de reconnaissance mots non contraints et omni scripteurs et sont retranscrits dans ce papier

Abstract *– This article analyses the behavior of various hybrid architectures based on a neuro-markovian scheme (MS-TDNN HMM) applied to online handwriting word recognition systems. We have considered different cost functions, including maximal mutual information criteria with discriminant training and maximum likelihood estimation, to train the systems globally at the word level and also we varied the number of states from one up to three to model the basic hidden Markov models at the letter level. We report on experimental results for non constrained, writer independent, word recognition obtained on the IRONOFF database.*

1. Introduction

Nos travaux s'intègrent dans le contexte de la reconnaissance de l'écriture en ligne destiné aux systèmes mobiles communicants (assistant numérique personnel, ardoise électronique, smart-phone). Dans ce domaine, les systèmes de reconnaissance mot les plus contraints imposent une écriture de type script [1, 2] facilitant ainsi le problème de segmentation. Les systèmes supportant une écriture cursive sans contrainte conduisent à une architecture nettement plus complexe, c'est précisément le cas du système présenté ici. Nous avons opté pour un système hybride neuromarkovien avec une architecture de type TDNN (Time Delay Neural Network) pour le réseau de neurones. Si de tels systèmes ont déjà été abordés dans la littérature [3, 4, 5], leurs apprentissages posent encore de nombreux problèmes. Nous proposons ici une méthode d'apprentissage global au niveau mot basée sur une fonction de coût générique qui permet de mixer apprentissage de type discriminant et apprentissage de type modèle générateur. La solution proposée représente une simplification intéressante par rapport aux approches nécessitant plusieurs étapes, quelques fois alternées, pour mener à bien l'apprentissage du niveau graphème ou lettre au niveau mot.

Pour illustrer ces travaux, nous décrirons dans un premier temps le processus de reconnaissance du système hybride neuro-markovien (section 2). En section 3, nous détaillons les différents modes d'apprentissage obtenus par variation des paramètres de la fonction de coût présenté dans ce papier avec ces résultats en quatrième partie. Pour améliorer notre système de reconnaissance mot nous présentons en section 5 une première solution, soit une extension multi états de la modélisation markovienne au niveau lettre.

2. Architecture du système

La figure 1 illustre le système de reconnaissance global [6]. Il est basé sur une approche analytique avec segmentation implicite et un apprentissage global au niveau mot.

De plus, il permet de manipuler un lexique de taille variable et ne requiert aucun apprentissage additionnel pour ajouter de nouvelles entrées au dictionnaire mot.

La première étape consiste à normaliser le signal d'écriture, principalement en taille, orientation des lignes de base et s'affranchir de la vitesse d'écriture. Une fois le signal normalisé, sept caractéristiques sont extraites en chaque point, elles seront présentées en entrée du système TDNN+HMM. Le TDNN balaie régulièrement la trajectoire du signal d'écriture avec un pas de segmentation implicite et une fenêtre d'une largeur donnée, elle-même balayée par une fenêtre de convolution de taille fixée à la hauteur du corps du mot. Pour chaque position du TDNN, celui-ci calcule un vecteur O_t correspondant aux probabilités a posteriori des classes associées à ses différentes sorties (66 classes : 2*26 lettres + 14 extensions), elles représentent les probabilités d'observations du HMM. Le HMM [7] effectue la tâche de reconnaissance au niveau mot en trouvant le meilleur

alignement temporel (programmation dynamique) sur les différents modèles mots du lexique.

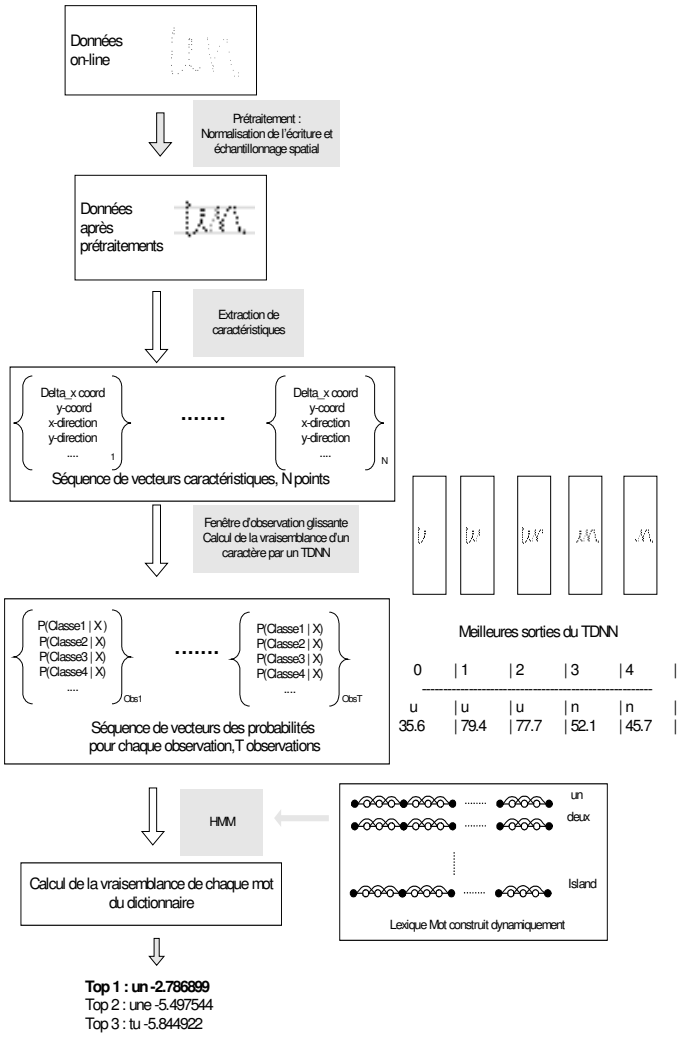

FIG 1 : Processus de reconnaissance du système TDNN+HMM (cas du mot « un »)

Avec une telle approche, une segmentation du mot en lettres est obtenue à l'issue de la reconnaissance, elle correspond aux changements de modèles-lettres sur le meilleur chemin du treillis de reconnaissance du modèle-mot considéré (Viterbi). Le mécanisme d'apprentissage associé à ce système tire profit de l'étiquetage des observations issues du TDNN résultant de cette segmentation implicite. Celui-ci est décrit dans la section suivante.

3. Apprentissages discriminants au niveau mot

L'objectif ici est de rétro-propager directement l'erreur au niveau mot dans le réseau de neurones pour mettre à jour ses paramètres en se basant sur l'étiquetage implicite. Pour cela, nous avons construit une fonction de coût générique L_G combinant à la fois un critère MLE (Maximum Likehood) et un critère MMI (Maximum Mutual Information) permettant de discriminer le modèle mot HMM reconnu à partir d'un lexique et/ou les sorties du TDNN mal classées.

$$L_G = (1+\varepsilon)\ \log P(O|\lambda_{HMMvrai}) - \beta.\begin{bmatrix}(1-\alpha)\log P(O|\lambda_{HMMreconnu}) \\ +\alpha\ \log P(O|\lambda_{TDNNreconnu})\end{bmatrix} \quad (1)$$

$$avec\ HMMreconnu = \arg\max_i P(O|\lambda_i)$$

Les paramètres α, β, et ε sont compris entre 0 et 1, le tableau 1 illustre les variantes possibles du critère. Avec $\varepsilon=\beta=0$, nous obtenons le critère classique du maximum de vraisemblance (MLE). Avec $\beta = 1$, on introduit un apprentissage discriminant (MMI simplifié) qui prend en compte uniquement le mot reconnu en 1ère position par l'algorithme de Viterbi ($\lambda_{HMMreconnu}$) si $\alpha=0$, et dans le cas où $\alpha=1$ seulement les classes reconnues en 1ère position en sortie du TDNN ($\lambda_{TDNNreconnu}$). La valeur α permet de pondérer ces deux derniers types de discrimination.

TAB. 1 : Paramètres en fonction des critères

Paramètres/Critères	ε	β	α
(1) MLE_HMM	0	0	0
(2) MMI_HMM	0	1	0
(3) MLE + MMI_HMM	1	1	0
(4) MLE + MMI_TDNN	1	1	1
(5) Mixte	1	1	0,5

Par souci de place, nous ne détaillerons pas les calculs permettant d'aboutir au gradient de l'erreur L_G donné ci-dessous pour la couche de sortie du TDNN, les couches cachées suivant l'algorithme de rétro propagation classique [7] :

$$\frac{\partial L_G}{\partial w_{ji}} = \sum_t \delta_{j,t} \cdot x_i(O_t)$$

$$avec\ \delta_{j,t} = Grad_{j,t} - x_{j,t}\sum_k Grad_{k,t}$$

$$Grad_{j,t} = \begin{pmatrix}(1+\varepsilon)\dfrac{P(O,q_t=j|\lambda_{HMMvrai})}{P(O,|\lambda_{HMMvrai})} \\ -\beta.\begin{bmatrix}(1-\alpha)\dfrac{P(O,q_t=j|\lambda_{HMMreconnu})}{P(O,|\lambda_{HMMreconnu})} \\ +\alpha\dfrac{P(O,q_t=j|\lambda_{TDNNreconnu})}{P(O,|\lambda_{TDNNreconnu})}\end{bmatrix}\end{pmatrix} \quad (2)$$

0ù j est la position du neurone considéré, i celle du neurone associé de la couche inférieure, t l'indice temporel de l'observation et $x_j(O_t)$ *la sortie du neurone j* de la couche de sortie avec $x_j(O_t) = b_j(O_t)$ selon la notation courante des HMMs : $\lambda(A,B,\pi)$ [8]. $P(O,q_t=j|\lambda)$ est calculé par programmation dynamique [9]. Ainsi pour chaque observation O_t, un gradient positif est rétropropagé pour le vrai modèle HMM (dont une fraction ε correspond à un pur critère ML) et un gradient négatif pour le modèle HMM reconnu et /ou pour les classes reconnues par le TDNN.

TAB. 2 : Calcul du gradient selon les critères

x(j,t) = HMM Vrai (j,t)	x(j,t) = HMM reconnu (j,t)	x(j,t) = TDNN reconnu (j,t)	$Grad_{j,t}$ MLE (1) (ε=0, β=0)	$Grad_{j,t}$ MMI (2) (ε=0, β=1)	$Grad_{j,t\,t}$ Generic (5)
F(Faux)	F	F	*0*	*0*	*0*
F	F	V(vrai)	*0*	*–α*	*–βα*
F	V	F	*0*	*–(1–α)*	*–β(1–α)*
F	V	V	*0*	*–1*	*–β*
V	F	F	*1*	*1*	*1+ε*
V	F	V	*1*	*1–α*	*1+ε–βα*
V	V	V	*1*	*1–(1–α)*	*1+ε –β(1–α)*
V	V	V	*1*	*0*	*1+ε–β*

Le tableau 2 illustre pour différents critères les valeurs prises par la variable $Grad_{j,t}$ en fonction de la sortie du réseau *x(j,t)* et de son appartenance aux différents chemins : faux (F) signifie que le chemin ne passe par la sortie j du réseau en cette observation t et vrai (V) le contraire. Le HMM vrai correspond au chemin Viterbi du modèle vrai connu, le HMM reconnu au chemin du mot du lexique avec la plus grande vraisemblance et le TDNN reconnu aux sorties de plus forte probabilité. Une normalisation du gradient entre -1 et 1 est ensuite réalisée pour assurer la convergence du réseau.

4. Expérimentations des critères d'apprentissage

Nous avons testé différentes combinaisons du critère précédent sur la base IRONOFF [10]. Un sous-ensemble de 20 898 mots, représentant 197 labels différents, est utilisé pour l'apprentissage et un ensemble séparé de 10 448 mots est testé en généralisation. Le tableau 3 donne les taux de reconnaissance en première position de la base IRONOFF pour chaque critère donné au tableau 1.

TAB. 3 : Taux de reconnaissance sur la base de généralisation (non apprise) IRONOFF

Critères	(1)	(2)	(3)	(4)	(5)
Taux	77,43	83.82	86,34	82,26	87.09

Ce tableau montre qu'un apprentissage uniquement avec un critère de type MLE (1) donne des résultats assez médiocres (77.4 %). Cela doit être mis en relation avec la taille de la base d'apprentissage qui ne permet pas une modélisation suffisamment précise de chaque classe. Dans ces conditions, faire intervenir une compétition entre les classes, avec un critère de discrimination simple (2) où le gradient est nul si le modèle vrai correspond au modèle reconnu permet d'améliorer sensiblement le taux de reconnaissance (83.8 %). La combinaison des deux précédents critères (3) apporte encore un bénéfice notable sur le taux de reconnaissance (86.3 %).

Enfin, un point important apparaît : une discrimination au niveau caractère (α=0.5) en plus d'une discrimination au niveau mot (5) accroît les performances du reconnaisseur mot. Il est intéressant de souligner la convergence constatée expérimentalement de ce système d'apprentissage en dehors de toute initialisation spécifique du TDNN et de toute labellisation explicite au niveau caractère. Toutefois, mais c'est le cas assez systématiquement avec les approches neuronales, la valeur du paramètre de pas de gradient influence de manière importante la qualité de la convergence.

5. Modélisation markovienne

Les expériences précédentes étaient fondées sur une modélisation simplifiée : une lettre était représentée par un HMM à un état avec des probabilités de transitions équiprobables et un mot résulte de la concaténation de modèles HMM lettres. Nous avons cherché à assouplir ce modèle, en utilisant des modèles lettres multi-états avec des transitions équiprobables, la figure 2 illustre les modèles lettres à 1 et 3 états. Cette modification a bien sûr une incidence sur la topologie de la couche de sortie du TDNN, celle-ci comporte autant de sorties que d'états distincts. Pour un modèle 3-états, cela fait donc 3 × 66 = 198 sorties.

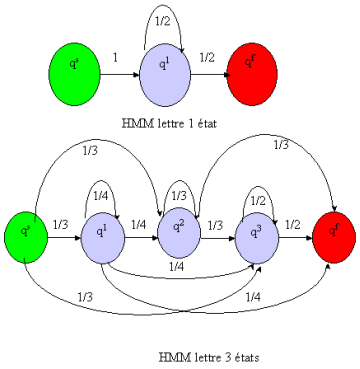

FIG 2 : Modélisation des HMM lettres 1 et 3 états

Le tableau 4 montre l'intérêt d'étendre le nombre d'états du modèle lettre, et cela quelque soit le critère choisi. Par exemple, dans le cas d'un critère MLE-MMI, le modèle 3 états permet de réduire l'erreur de reconnaissance de 41 pourcents. Deux raisons claires sont la conséquence de ces

résultats. La première réside dans les prétraitements du signal d'écriture et la topologie du réseau. La fenêtre d'observation est égale à 4/3 de la longueur moyenne d'un caractère, elle est décalée d'une observation à l'autre de 1/3 de cette longueur. Ainsi un même caractère participe à plusieurs observations consécutives. Avec un seul état il est plus difficile de rendre compte des bruits générés par les lettres qui l'entourent et par les ligatures. Avec plusieurs états il est plus facile d'absorber les différentes situations du caractère dans la fenêtre (début, milieu et fin de la fenêtre). La seconde raison est la plus grande souplesse du modèle HMM pour intégrer les variabilités de longueur intra et inter-lettres. Contenu du nombre variable de graphèmes dans une lettre, une lettre courte à un seul graphème (comme le « i », le « c ») pourra ainsi traverser le modèle par un seul état tandis qu'une lettre longue pourra s'étaler sur deux ou trois états (cas du « f », du « m », du « w »).

TAB. 4 : Taux de reconnaissance du système multi-états

Critères	(1)	(2)	(3)	(5)
1 état	77,43	83,82	86,34	87,09
2 états	80,87	87.46	88.22	ND
3 états	84,69	90,57	92,01	92,78

(ND : non disponible actuellement).

Les approches neuro-markoviennes permettent d'introduire un apprentissage discriminant local et apportent une première réponse intéressante au problème de la discrimination (MMI, combinaison MMI-MLE). L'introduction d'un critère plus local telle une discrimination des sorties du TDNN par rapport au modèle vrai donné par Viterbi permet de renforcer l'apprentissage tant dans sa phase d'initialisation qu'en terme de reconnaissance.

6. Conclusion et perspectives

Dans ce papier nous avons étudié différents modes d'apprentissage global d'un reconnaisseur mot en-ligne basé sur une architecture TDNN-HMM. Nous avons établi une fonction générique qui permet en fonction de ses paramètres de jouer avec des critères plutôt discriminants MMI ou des critères classiques MLE. La combinaison de ces critères est très intéressante en terme de taux de reconnaissance. Il est important de souligner qu'aucune phase d'apprentissage sur des sous-entités segmentées à la main n'a été réalisée, contrairement à ce qui est souvent le cas dans les schémas actuels dans la littérature.

Plusieurs améliorations peuvent être apportées à ce système. Il pourrait être envisagé d'optimiser le nombre d'états lettre par lettre, de travailler sur les modèles de durée et cela de façon dynamique au cours de l'apprentissage afin d'introduire d'abord un *a-priori* assez fort, et également de faire évoluer la fonction de coût générique, en jouant au fur et à mesure des itérations sur les paramètres de contrôle.

Références

[1] L. Oudot, L. Prevost L. & M. Milgram. *Un modèle d'activation vérification pour la lecture de textes manuscrits dynamique*. Colloque International Francophone sur l'écrit et le Document (*CIFED'04)*, La Rochelle, France, 2004.

[2] N. Ragot, E. Anquetil. *A generic hybrid classifier based on hierarchical fuzzy modeling experiments on on-line handwritten character recognition*. Seventh International Conference on Document Analysis and Recognition *(ICDAR'03),* UK, vol. 2, pages 963-967, 2003.

[3] M. Schenkel, I. Guyon, D. Henderson. *On-line cursive script recognition using Time Delay Neural Networks and Hidden Markov Models*. Machine Vision and Applications, special issue on Cursive Script Recognition, (8):215--223, 1995.

[4] S. Jaeger, S. Manke, J. Reichert, A. Waibel. *On-Line Handwriting Recognition: The NPen++ Recognizer*. International Journal on Document Analysis and Recognition *(IJDAR'00)*, volume 3, pages 169-180, 2000.

[5] Z. Wimmer, S. Garcia-Salicetti, A. Lifchitz, B. Dorizzi, P. Gallinari, T. Artières. « REMUS », sur la toile http://www-connex.lip6.fr/~lifchitz/Remus/ .

[6] E. Poisson, C. Viard-Gaudin, P.-M. Lallican. *Système TDNN/HMM de reconnaissance de mots cursifs en ligne à apprentissage simplifié*. Colloque International Francophone sur l'écrit et le Document (*CIFED'04),* La Rochelle, Juin 2004.

[7] Y. LeCun, L. Bottou., Y. Bengio , P. Haffner. *Gradient-Based Learning Applied to Document Recognition*. Intelligent Signal Processing, pages 306-351, 2001.

[8] L.R. Rabiner. *A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition*. Proceedings of IEEE, volume 77, pages 257-285, 1989

[9] R. S. Sutton and A.G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998, A Bradford Book.

[10] C. Viard-Gaudin, P.M. Lallican, S. Knerr, P. Binter. *The IRONOFF Dual Handwriting Database*. International Conference on Document Analysis and Recognition *(ICDAR'99)*, pages 455-458. Bangalore, Sept. 1999.