Classification probabiliste non supervisÈe et visualisation des donnÈes sÈquentielles

Rakia Jaziri [*,**], Mustapha Lebbah[*], YounËs Bennani[*],

[*]LIPN-UMR 7030 - CNRS, Université Paris 13,
99, av. J-B Clément F-93430 Villetaneuse
Prénom.Nom@lipn.univ-paris13.fr
[**]Institut National de l'Audiovisuel,
4, av. de l'Europe 94 366 Cedex Bry-sur-Marne
rjaziri@ina.fr

Résumé. Nous proposons dans ce papier un nouvel algorithme de classification non supervisée à base de modèle de mélange topologique pour des données non i.i.d (non independently and identically distributed). Ce nouveau paradigme probabiliste, plonge les cartes topologiques probabilistes dans une formulation sous forme de chaînes de Markov cachées. Dans cette formulation, la génération d'une observation à un instant donné du temps est conditionnée par les états voisins au même instant du temps. Ainsi, une grande proximité impliquera une grande probabilité pour la contribution à la génération. L'approche proposée est évaluée en utilisant des données séquentielles réelles issues des bases de données de l'Institut Nationale de l'Audiovisuel (INA). Les résultats obtenus sont très encourageants et prometteurs.

1 Introduction

Plusieurs techniques de classification automatique des données séquentielles ont été développées ces dernières années. Elles ont été appliquées dans différents domaines tels que la reconnaissance des caractères manuscrits (Prat et al., 2009), la reconnaissance de la parole, l'étude de la mobilité des objets dans les vidéos (Buzan et al., 2004) et l'analyse de séquences biologiques (ADN). La méthode la plus facile pour traiter ce type de données serait tout simplement d'ignorer l'aspect temporel et de traiter les observations comme des données indépendantes ou i.i.d "independent and identically distributed". Pour beaucoup d'applications, l'hypothèse i.i.d rend les données plus pauvres en perdant l'information séquentielle. Souvent dans beaucoup d'applications le traitement est décomposé en deux étapes : la première est l'étape de classification ou de partitionnement des données avec l'hypothèse i.i.d. Dans la deuxième étape, le résultat de la classification est utilisé pour construire un modèle probabiliste en relaxant la contrainte i.i.d, et une des plus naturelles manières de faire cela est d'utiliser un modèle de Markov.

Les cartes topologiques (Kohonen, 2001) sont intéressantes de par leurs apports topologiques à la classification non supervisée et leurs capacités à résumer de manière simple un

ensemble de données multi-dimensionnelles. Elles permettent d'une part de comprimer de grandes quantités de données en regroupant les individus similaires en classes, et d'autre part de projeter les classes obtenues de façon non linéaire sur une carte ou un graphe. Ce procédé permet donc d'effectuer une réduction de dimension, permettant ainsi de visualiser la structure des données en deux dimensions tout en respectant la topologie des données, de sorte que deux données proches dans l'espace multi-dimensionnel de départ aient des images proches sur la carte. Dans la plupart des modèles hybrides, les cartes topologiques sont utilisées comme un pré-traitement pour la quantification vectorielle et les HMMs (Hidden Markov Models) sont ensuite utilisés dans des processus de transformation plus avancés pour la prise en compte de la dynamique. Les résultats de classification dépendent essentiellement du choix des deux paramètres : la distribution de probabilité et la topologie de la carte d'auto-organisation. Malheureusement, les paradigmes de l'auto-organisation ne peuvent pas être facilement transférés à des données non i.i.d. Différentes approches ont été développées pour intégrer l'information temporelle dans la carte d'auto-organisation. Une variété de modèles existe pour les cartes auto-récurrentes : la carte de Kohonen temporelle (TKM), la SOM récurrente (RSOM), la SOM récursive (RecSOM), et SOM pour les données structurées (SOMSD), (Strickert et Hammer, 2003; Hagenbuchner et al., 2003).

Nous proposons dans cet article une nouvelle approche de carte topologique probabiliste dédiée aux données séquentielles multivariées, que nous appelons : carte auto-organisatrice probabiliste pour les données séquentielles (PrSOMS). Nous supposons que les données séquentielles sont générées selon un processus markovien. Les modèles de Markov cachés figurent parmi les meilleures approches adaptées aux traitements des séquences, étant donné leur capacité à traiter des séquences de longueurs variables et leur pouvoir à modéliser la dynamique d'un phénomène décrit par des suites d'événements. La modélisation graphique probabiliste motive différentes structures graphiques basées sur les HMMs (Bengio et Frasconi, 1994). Les modèles graphiques fournissent un formalisme général pour décrire et analyser de telles structures. Par conséquent, il est très important d'avoir des algorithmes capables de déduire à partir d'un ensemble de données de séquences non seulement la probabilité de distribution, mais aussi la structure topologique du modèle, c'est-à-dire, le nombre d'états et les transitions qui les inter-connectent. Malheureusement, cette tâche est très difficile et on a que des solutions partielles. Afin de surmonter les limites des HMMs, des travaux récents dans (Bouchaffra, 2008) qui proposent un nouveau et un original paradigme, appelé *topological HMM*, qui manipule les nœuds du graphe associé au HMM et ses transitions dans un espace Euclidien. Cette approche modélise la structure locale d'un HMM et extrait leur forme en définissant une unité d'information comme une forme composée d'un groupe de symboles d'une séquence. Un autre modèle, souvent présenté comme la version probabiliste de la carte d'auto-organisation nommé (GTM) qui a été étendu à un modèle de série chronologique univariées (GTM through time) (Bishop, 2006; Olier et Vellido, 2008) et aux données structurées (Bacciu et al., 2010). Récemment, dans (Yamaguchi, 2010), l'auteur propose l'extension de l'algorithme SOMM (Self-Organizing Mixture Model, (Verbeek et al., 2005)) pour les séries chronologiques multivariées (SOHMMs : self-organizing hidden Markov models). Cependant, la manière dont ces modèles réalisent l'organisation topographique est tout à fait différente de celles utilisées dans les algorithmes SOM et notre modèle (PrSOMS).

Dans cet article, nous nous sommes intéressés à la problématique d'analyse de données structurées en séquences, qu'elles soient de longueurs fixes ou variables. L'objectif de cette ap-

proche est de construire un nouveau modèle auto-organisé génératif d'un ensemble de données non i.i.d. Le modèle proposé est basé sur le formalisme probabiliste des cartes topologiques (Anouar et al., 1997; Lebbah et al., 2007), et le modèle génératif utilisé dans les HMM. Par conséquent, il consiste à estimer les paramètres du modèle en maximisant la vraisemblance de l'ensemble des données séquentielles. L'algorithme d'apprentissage que nous proposons est une application de l'algorithme standard EM "Espérance-Maximisation". Cet article est organisé comme suit. Les sections 2 et 3 présentent notre approche probabiliste pour la classification des données séquentielles(PrSOMS). La section 4 décrit le dispositif expérimental et les évaluations. Enfin, la section 5 propose une conclusion et des perspectives.

2 Principe du modèle génératif PrSOMS

Dans notre modèle, nous considérons une carte topologique formant une grille qui sera vu comme une chaîne de Markov (un HMM). La génération d'une variable observable à un instant donné du temps est conditionnée par les états voisins au même instant du temps. Ainsi, une grande proximité implique une grande probabilité pour la contribution de la génération. Cette proximité est quantifiée en utilisant la fonction de voisinage. Le formalisme que nous présentons est valable pour toutes les structures. Supposons une séquence d'observations $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n, ..., \mathbf{x}_N\}$ tel que $\mathbf{x}_n$ est un élément de la séquence de taille N. La principale problématique est d'estimer les paramètres du modèle d'apprentissage PrSOMS. On suppose que l'architecture de la carte modélisant aussi un HMM est représentée par un treillis $\mathcal{C}$, qui a une topologie discrète définie par un graphe non orienté. On notera le nombre des cellules (noeuds,états) de $\mathcal{C}$ par K. Pour chaque paire de cellules (c,r) dans le graphe, la distance $\delta(c, r)$ est définie comme la longueur de la plus courte chaîne qui lie les cellules ou les états c et r. On suppose que chaque élément $\mathbf{x}_n$ d'une séquence d'observations $\mathbf{X}$ est généré par le processus suivant : on commence par associer à chaque cellule (état) $c \in \mathcal{C}$ une probabilité $p(\mathbf{x}_n/c)$ où $\mathbf{x}_n$ est un vecteur dans l'espace des données. Par la suite, on sélectionne une cellule c^* de la carte $\mathcal{C}$ selon une probabilité a priori $p(c^*)$. Pour chaque cellule c^*, on sélectionne une cellule $c \in \mathcal{C}$ selon la probabilité conditionnelle $p(c/c^*)$. Toutes les cellules $c \in \mathcal{C}$ et au même instant n contribuent à la génération d'un élément $\mathbf{x}_n$ avec la probabilité $p(\mathbf{x}_n/c)$ selon la proximité à la cellule c^* décrite par la probabilité $p(c/c^*)$. Nous introduisons notamment deux variables binaires aléatoires comme variables cachées $\mathbf{z}_n$ et $\mathbf{z}_n^*$ de dimension K, dans lesquelles deux éléments particuliers z_{nr} et z_{nc}^* est égaux à 1 et tous les autres éléments sont égaux à 0. Les deux composantes z_{nc}^* et z_{nr} indiquent un couple d'états responsable de la génération d'un élément de l'observation. Utilisant cette notation on peut réécrire la probabilité $p(\mathbf{x}_n/c)$ comme suit :

$$p(\mathbf{x}_n/c) \simeq p(\mathbf{x}_n/z_{nc} = 1) \simeq p(\mathbf{x}_n/\mathbf{z}_n)$$

et $p(c/c^*) = p(z_{nc} = 1/z_{nc^*}^* = 1) \simeq p(z_{nc}/z_{nc^*}^*) \simeq p(\mathbf{z}_n/\mathbf{z}_n^*)$
Pour introduire le processus d'auto-organisation dans l'apprentissage du modèle de mélange on suppose que $p(z_{nc}/z_{nsc^*}^*)$ peut être définie de la même manière que les modèles des cartes probabilistes : $p(z_{nc}/z_{nc^*}^*) = \frac{\mathcal{K}^T(\delta(c,c^*))}{\sum_{r\in\mathcal{C}} \mathcal{K}^T(\delta(r,c^*))}$,
où $\mathcal{K}^T$ est la fonction de voisinage qui dépend du paramètre T (appelé température) : $\mathcal{K}^T(\delta) = \mathcal{K}(\delta/T)$. Ainsi $\mathcal{K}$ définit pour chaque état de la chaîne de Markov $z_{nc^*}^*$ une région de voisinage

dans le graphe $\mathcal{C}$. Le paramètre T permet de contrôler la taille du voisinage qui influence une cellule donnée de la carte $\mathcal{C}$. La valeur de T varie entre deux valeurs T_{max} et T_{min}.

On note l'ensemble de toutes les variables cachées par $\mathbf{Z}^*$ et $\mathbf{Z}$, où chaque ligne $\mathbf{z}_n^*$ et $\mathbf{z}_n$ est associé à chaque élément de la séquence $\mathbf{x}_n$. Chaque observation de la séquence en X, est associée à un couple de variables cachées $\mathbf{Z}$ et $\mathbf{Z}^*$ responsables de la génération. On note par $\{\mathbf{X}, \mathbf{Z}, \mathbf{Z}^*\}$ l'ensemble complet des données, et on se réfère aux données observables $\mathbf{X}$ comme incomplètes. Ainsi, le modèle générateur d'une séquence est défini de la manière suivante :

$$p(\mathbf{X}; \theta) = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}, \mathbf{Z}^*; \theta) \tag{1}$$

Puisque la distribution $p(\mathbf{X}, \mathbf{Z}, \mathbf{Z}^*; \theta)$ ne peut pas se simplifier, une caractéristique importante pour les distributions des probabilités sur des variables multiples est celle de l'indépendance conditionnelle (Luttrel, 1994). On suppose que la distribution conditionnelle de $\mathbf{X}$, sachant $\mathbf{Z}^*$ et $\mathbf{Z}$, ne dépend pas de la variable cachée $\mathbf{Z}^*$. Souvent cette hypothèse est utilisée pour les modèles graphiques, ainsi $p(\mathbf{X}/\mathbf{Z}, \mathbf{Z}^*) = p(\mathbf{X}/\mathbf{Z})$. On peut réécrire la distribution marginale comme :

$$p(\mathbf{X}; \theta) = \sum_{\mathbf{Z}^*} p(\mathbf{Z}^*) \sum_{\mathbf{Z}} p(\mathbf{Z}/\mathbf{Z}^*) p(\mathbf{X}/\mathbf{Z}) \tag{2}$$

avec

$$p(\mathbf{X}/\mathbf{Z}^*) = \sum_{\mathbf{Z}} p(\mathbf{Z}/\mathbf{Z}^*) p(\mathbf{X}/\mathbf{Z}) \tag{3}$$

3 Paramètres du modèle et estimation

Considérant que la carte $\mathcal{C}$ représente un modèle de Markov, ainsi la distribution à l'état $\mathbf{z}_n^*$ dépend de l'état de la variable latente précédente $\mathbf{z}_{n-1}^*$. Cette dépendance est représentée avec la probabilité conditionnelle $p(\mathbf{z}_n^*|\mathbf{z}_{n-1}^*)$. Puisque les variables latentes sont des variables binaires de dimension K, cette distribution conditionnelle correspond à une table de probabilité qu'on note par $\mathbf{A}$. Les éléments de $\mathbf{A}$ sont connus comme des probabilités de transition notées par :

$$A_{jk} = p(z_{nk}^* = 1/z_{n-1,j}^* = 1) \text{ avec } \sum_k A_{jk} = 1$$

On peut écrire la distribution conditionnelle explicitement sous cette forme

$$p(\mathbf{z}_n^*/\mathbf{z}_{n-1}^*, \mathbf{A}) = \prod_{k=1}^{K} \prod_{j=1}^{K} A_{jk}^{z_{n-1,j}^* z_{nk}^*}$$

L'état initial $\mathbf{z}_1^*$ est un cas particulier puisqu'il n'a pas d'état ou de cellule parente, et ainsi il a une distribution marginale $p(\mathbf{z}_1^*)$ représentée par un vecteur de probabilités π avec les éléments $\pi_k = p(\mathbf{z}_{1k}^* = 1)$, ainsi que $p(\mathbf{z}_1^*|\pi) = \prod_{k=1}^{K} \pi^{z_{1k}^*}$, où $\sum_k \pi_k = 1$.

Les paramètres du modèle sont complétés en définissant les distributions conditionnelles des variables observées $p(\mathbf{x}_n/\mathbf{z}_n; \phi)$, où ϕ est un ensemble de paramètres qui définissent la distribution qui est connue comme des probabilités d'émission dans le modèle HMM. On peut représenter les probabilités d'émission sous la forme : $p(\mathbf{x}_n/\mathbf{z}_n; \phi) = \prod_{k=1}^{K} p(\mathbf{x}_n; \phi_k)^{z_{nk}}$.

La probabilité jointe des variables observables et les deux variables latentes $\mathbf{Z}$ et $\mathbf{Z}^*$ est exprimée par :

$$p(\mathbf{X}, \mathbf{Z}^*, \mathbf{Z}; \theta) = p(\mathbf{Z}^*; \mathbf{A}) \times p(\mathbf{Z}/\mathbf{Z}^*) \times p(\mathbf{X}/\mathbf{Z}; \phi)$$

$$p(\mathbf{X}, \mathbf{Z}^*, \mathbf{Z}; \theta) = \left[p(\mathbf{z}_1^*|\pi) \prod_{n=2}^{N} p(\mathbf{z}_n^*/\mathbf{z}_{n-1}^*; \mathbf{A}) \right] \times \left[\prod_{i=1}^{N} p(\mathbf{z}_i/\mathbf{z}_i^*) \right] \times \left[\prod_{m=1}^{N} p(\mathbf{x}_m/\mathbf{z}_m; \phi) \right]$$

où $\theta = \{\pi, \mathbf{A}, \phi\}$ décrit l'ensemble des paramètres qui manipulent le modèle. Il n'est pas évident de maximiser la fonction de vraisemblance, à cause de la complexité de l'expression. C'est pour cela qu'on utilise l'algorithme EM pour trouver les paramètres qui maximisent la fonction de vraisemblance. L'algorithme EM commence avec quelques sélections initiales pour les paramètres du modèle, qu'on note par θ^{old}. Dans l'étape E (Estimation), on prend les valeurs des paramètres et on trouve la distribution a posteriori des variables latentes $p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}, \theta^{old})$. Ensuite on utilise cette distribution a posteriori pour évaluer l'espérance du logarithme de la vraisemblance des séquences complètes des données (eq.3), en fonction des paramètres θ, pour obtenir la fonction objective $Q(\theta, \theta^{old})$ définie par :

$$Q(\theta, \theta^{old}) = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}; \theta^{old}) \ln p(\mathbf{X}, \mathbf{Z}^*, \mathbf{Z}; \theta)$$

Ainsi on peut réécrire la fonction :

$$Q(\theta, \theta^{old}) = Q_1(\pi, \theta^{old}) + Q_2(\mathbf{A}, \theta^{old}) + Q_3(\phi, \theta^{old}) + Q_4 \tag{4}$$

oú

$$Q_1(\pi, \theta^{old}) = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} \sum_{k=1}^{K} p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}; \theta^{old}) z_{1k}^* \ln \pi_k$$

$$Q_2(\mathbf{A}, \theta^{old}) = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} \sum_{n=2}^{N} \sum_{k=1}^{K} \sum_{j=1}^{K} p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}; \theta^{old}) z_{n-1,j}^* z_n^* \ln(A_{jk})$$

$$Q_3(\phi, \theta^{old}) = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} \sum_{n=1}^{N} \sum_{k=1}^{K} p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}; \theta^{old}) z_{nk} \ln \left(p(\mathbf{x}_n; \phi_k) \right)$$

$$Q_4 = \sum_{\mathbf{Z}^*} \sum_{\mathbf{Z}} p(\mathbf{Z}^*, \mathbf{Z}/\mathbf{X}; \theta^{old}) \ln p(\mathbf{Z}/\mathbf{Z}^*)$$

A cette étape, on va introduire quelques notations. On va utiliser $\gamma(\mathbf{z}_n^*, \mathbf{z}_n)$ pour noter la distribution marginale *a posteriori* des variables latentes $\mathbf{z}_n^*$ et $\mathbf{z}_n$, et
$\xi(\mathbf{z}_{n-1}^*, \mathbf{z}_n^*) = p(\mathbf{z}_{n-1}^*, \mathbf{z}_n^*/\mathbf{X}, \theta^{old})$ pour noter la distribution a posteriori jointe des variables latentes successives, tel que :

$$\gamma(\mathbf{z}_n^*) = \sum_{\mathbf{z}} p(\mathbf{z}_n^*, \mathbf{z}_n|\mathbf{X}; \theta^{old}) \text{ et } \gamma(\mathbf{z}_n) = \sum_{\mathbf{z}^*} p(\mathbf{z}_n^*, \mathbf{z}_n|\mathbf{X}; \theta^{old})$$

On observe que la fonction objective (eq.4) $Q(\theta, \theta^{old})$ est définie comme une somme de quatre termes. Le premier terme $Q_1(\pi, \theta^{old})$ dépend des probabilités initiales ; le deuxième terme $Q_2(\mathbf{A}, \theta^{old})$ dépend des probabilités de transition $\mathbf{A}$; le troisième terme $Q_3(\phi, \theta^{old})$ dépend de ϕ qui est l'ensemble des paramètres de la probabilité d'émission, et le quatrième est une constante. La maximisation de $Q(\theta, \theta^{old})$ par rapport à $\theta = \{\pi, \mathbf{A}, \phi\}$ peut être effectuée séparément.

1. **La maximisation de $Q_1(\pi, \theta^{old})$: Les probabilités initiales**
   De la même manière que les modèles probabilistes, on utilise une forme explicite de la distribution des probabilités initiales. La probabilité initiale π est ensuite obtenue de la manière suivante : $\pi_k = \frac{\gamma(z^*_{1k})}{\sum_{j=1}^{K} \gamma(z^*_{1j})}$
2. **La maximisation de $Q_2(\mathbf{A}, \theta^{old})$: Probabilités de transition**
   Comme dans le cas des HMMs traditionnels, notre modèle utilise un état caché de valeur discrète avec une distribution multinomiale sachant les valeurs précédentes de l'état. Donc notre modèle est un modèle du premier ordre. La mise à jour des paramètres est calculée de la manière suivante :
   $$A_{jk} = \frac{\sum_{n=2}^{N} \xi(z^*_{n-1,j}, z^{*k}_n)}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z^*_{n-1,j}, z^*_{nl})} \quad (5)$$
   où
   $$\xi(z^*_{n-1,j}, z^*_{n,k}) = \mathbf{E}[z^*_{n-1,j} z^{*k}_n] = \sum_{\mathbf{z}^*} \gamma(\mathbf{z}^*) z^*_{n-1,j} z^*_{n,k}$$
3. **La maximisation de $Q_3(\phi, \theta^{old})$: Les probabilités d'émission**
   L'ensemble des paramètres ϕ dépend de la distribution utilisée. Nous présentons l'application en utilisant la loi gaussienne. Dans le cas des probabilités d'émission avec une densité sphérique gaussienne on a $p(\mathbf{x}/\phi_k) = \mathcal{N}(\mathbf{x}; \mathbf{w}_k, \sigma_k)$, définie par sa "moyenne" $\mathbf{w}_k$, qui a la même dimension que les données d'entrée, et sa matrice de covariance, définie par $\sigma_k^2 \mathbf{I}$ où σ_k est l'écart-type et $\mathbf{I}$ est la matrice identité,
   $$N(\mathbf{x}; \mathbf{w}_k, \sigma_k) = \frac{1}{(2\pi\sigma_k)^{\frac{d}{2}}} \exp\left[\frac{||\mathbf{x} - \mathbf{w}_k||^2}{2\sigma_k^2}\right]$$
   La maximisation de la fonction $Q_3(\phi, \theta^{old})$ fournit les expressions connues :
   $$\mathbf{w}_k = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n}{\sum_{n=1}^{N} \gamma(z_{nk})} \quad (6)$$
   $$\sigma_k^2 = \frac{\sum_{n=1}^{N} \gamma(z_{nk}) ||\mathbf{x}_n - \mathbf{w}_k||^2}{d \sum_{n=1}^{N} \gamma(z_{nk})} \quad (7)$$
   où d est la dimension de l'élément $\mathbf{x}$.

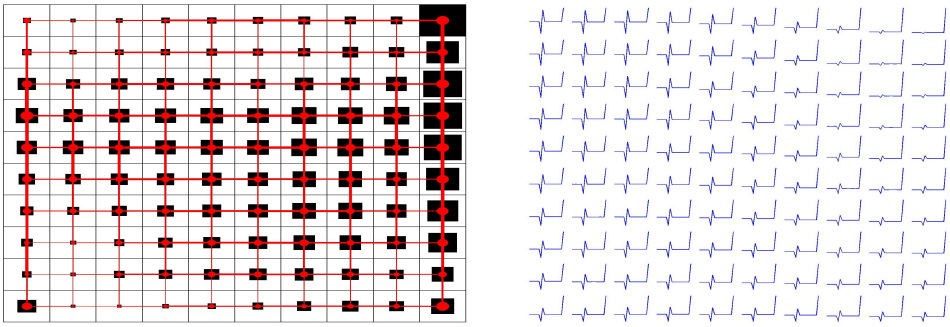
Dans le contexte particulier du modèle de Markov caché, on va utiliser l'algorithme forward-backward (Rabiner, 1989), puisqu'on utilise la structure du graphe pour organiser les données séquentielles d'une manière explicite. Quelques formules sont similaires aux chaînes de Markov traditionnelles si on n'utilise pas la structure du graphe.

4 Expérimentations

L'analyse des séquences audiovisuels peut être vu comme un problème d'analyse des séquences multidimensionnelles. Dans cette partie, nous allons discuter notre approche sur une base de données réelles issue de l'INA. Nous l'avons appliqué aussi sur d'autres bases publiques, mais vu la taille de l'article nous nous sommes contenté de l'application réelle de l'INA. La validation est réalisée à l'aide de l'expert du domaine, un croisement avec le guide du programme et la visualisation des segments audiovisuels. Les données se composent de 10 séquences de longueurs variables (10 chaines TV). Chaque séquence représente les différents segments diffusés dans une chaine de télévision pendant une journée. Chaque segment est une composante multidimensionnelle caractérisée par 19 variables. Ces variables sont validées par un expert du domaine à l'INA. Elles contiennent des descripteurs sur le nombre de répétitions, l'intervalle, la durée et d'autres variables que nous ne pouvons pas divulguer dans ce papier. Nous rappelons ici que le problème est de détecter automatiquement les segments de la séquence qui sont homogènes d'une part, et de reconstruire des séquences par la réunification des segments préalablement identifiés comme segment du programme d'une autre part. Nous avons commencé par appliquer notre approche PrSOMS sur les données. La séquence la plus probable est obtenue en utilisant l'algorithme de Viterbi (Rabiner, 1989). Ainsi il nous permet d'affecter chaque élément de la séquence.

Nous présentons dans ce qui suit l'expérimentation qui évalue le pouvoir de structuration de notre approche. La figure 1(a) schématise la carte PrSOMS (10×10) des états latents qui sont représentés par des carrés mis à l'échelle en fonction de la cardinalité des éléments de toutes les séquences du modèle capturées ou affectées à l'aide de l'algorithme de Viterbi. Les lignes rouges représentent le chemin de Viterbi pour chaque séquence. La largeur des lignes reflète le nombre de transition entre les états. La figure 1(b) schématise les profils ou les prototypes associés à chaque état (le centre de la distribution gaussienne). Chaque cellule est représentée par un vecteur de 19 composantes. La carte topologique PrSOMS nous permet de visualiser la partition. Ainsi, les experts du domaine pourraient utiliser ces cellules pour analyser certaines caractéristiques du flux audiovisuel. En effet, en faisant une analyse visuelle des segments vidéo de chaque cellule, nous avons pu tirer des caractéristiques propres à chaque chaine et qui les fait distinguer des autres. Ce qui est utile pour les experts du domaine. Les deux figures permettent d'analyser toutes les séquences de 10 chaînes en même temps.

En effet, la distribution de deux séquences différentes sur la carte PrSOMS devrait être sensiblement différente. Nous allons illustrer cela avec une comparaison entre les chemins de Viterbi donnés par deux chaines de télévision différentes, en se basant sur l'illustration des principales différences dans la visualisation de séquences. Les figures 2(a) et 2(b) affichent respectivement la carte représentant le chemin de Viterbi calculé avec la chaine 1 et la chaine 2. La représentation correspondant à la chaine 1 (figure 2 (a)) est plus compact que celle de la chaine 2 (figure 2(b)) étant donnée que les séquences de la chaine 1 sont moins variables que celle de la chaine 2. La chaine 1 est plus concentrée dans la partie inférieure à gauche et en haut à droite de la carte. Après analyse la chaine1 montre un comportement d'une chaine généralise et chaine 2 le comportement d'une chaine d'information. Les figures 2(c) et 2(d) montrent le chemin de Viterbi de deux sous séquences passant par les mêmes états afin d'identifier la similarité entre les séquences. Les figures 3 et 4 affichent respectivement des images extraites des deux cellules qui sont en haut à droite et en bas à gauche. Nous constatons que la majorité des vidéos capturées correspondent aux programmes courts tel que (météo, nouvelles brèves,



(a) (b)

FIG. 1 – *(a) Carte PrSOMS ; les carrés et les lignes indiquent respectivement la cardinalité des cellules et les transitions capturées en utilisant l'algorithme de Viterbi. (b) Les profils associés à chaque cellule ou état.*

publicités) qui est une caractéristique de la chaine 2. La figure 4 affiche des films capturés par l'état qui est en bas à gauche dans la chaine 1 (chaîne généraliste). Notre approche détecte la structure sous-jacente du flux et les résultats montrent une précision temporelle conforme à la vérité terrain et bien meilleure que celle indiquée dans les guides de programmes TV fournis par les chaînes de télévision. Afin de se rapprocher des partitionnement d'experts, nous avons appliqué la classification ascendante hiérarchique (CAH) sur les états ou cellules de la carte PrSOMS pour obtenir 8 clusters, et ceci en relaxant la contrainte non-i.i.d. La figure 6 montre la carte PrSOMS segmentée par une CAH. Tout d'abord, nous remarquons lors de notre expérimentations que plus de 90 % des inter-programmes (Publicités, Jingle, météos) sont diffusés au moins deux fois dans la journée dans plus de dix chaines de télévisions. Deuxièmement, seuls environ 30 % des programmes courts sont répétés. La précision des programmes extraits a également été évaluée. Le début (respectivement à la fin) de chaque programme a été extrait par rapport au démarrage effectif (respectivement à la fin) donné par des observations réelles.

Dans la suite, nous visualisons pour chaque cluster de la carte PrSOMS un ensemble de prototypes d'images composantes. Dans la figure 5, nous avons remarqué que les clusters sont composés de contenu homogène. En effet, il existe des clusters avec uniquement "émissions" ou d'autres où il n'y a que des "publicités". Nous avons constaté aussi que les segments répétés se regroupent ensemble ou dans des cellules voisines. Nous remarquons aussi que les segments d'inter-programme ont été classés suivant leur catégorie : publicité, bande annonce, parrainage, jingle, etc. Dans notre cas, la classification automatique est parfaitement conforme à nos observations réelles, alors que les indicateurs du guide sont assez décalés. Les résultats sont très encourageant. Une difficulté majeure est que la qualité des résultats dépend de la détection de répétitions et du nombre de variables utilisées pour le clustering. Les résultats obtenus montrent que notre approche a effectué une bonne segmentation du flux TV. L'analyse des données montre que la plupart des inter-programmes sont diffusés plusieurs fois. Ces programmes courts partagent de nombreuse caractéristiques, ce qui perturbe notre approche.

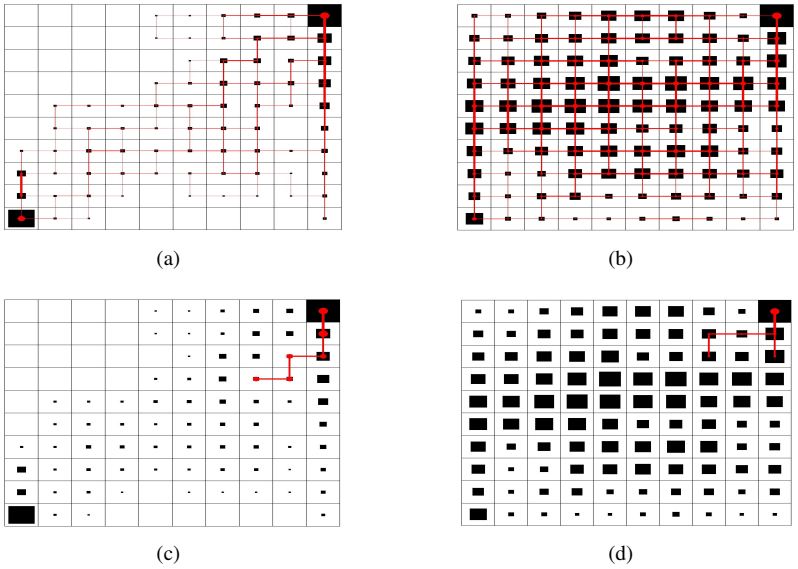


FIG. 2 – *(a) chaine 1. (b) chaine 2. (c) Sous-séquence de la chaine1. (d) Sous-séquence de la chaine 2.*

FIG. 3 – *Images des vidéos capturées en haut à droite de la carte*

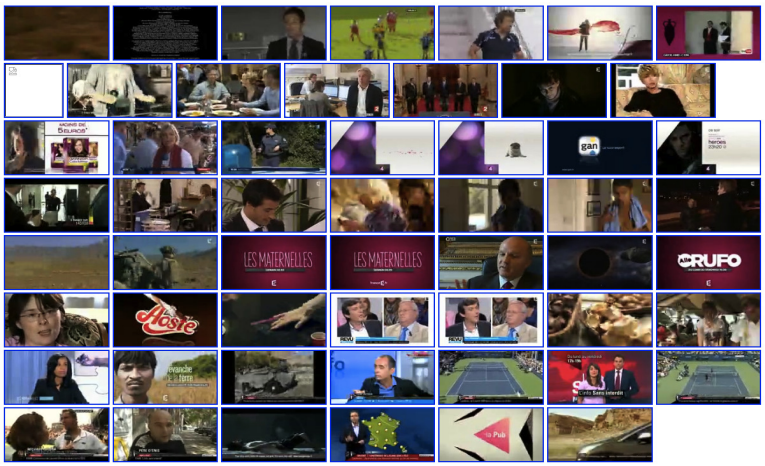


FIG. 4 – *Images des vidéos capturées sur le coin inférieur gauche de la carte*

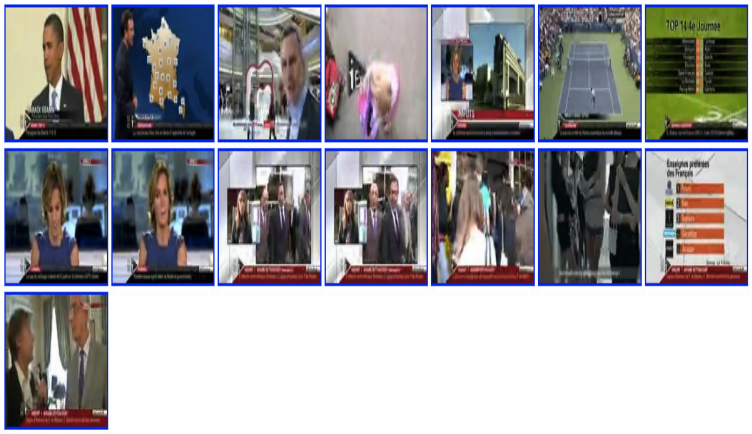


FIG. 5 – *Quelques images décrivant chaque cluster.*

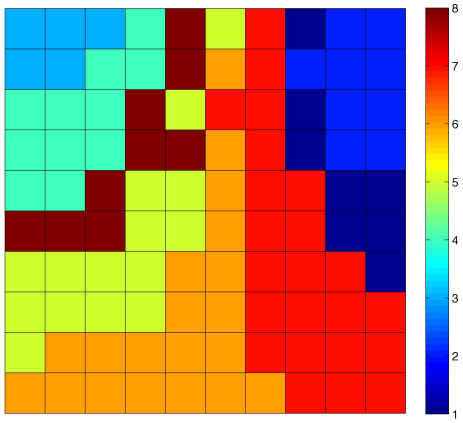


FIG. 6 – *La carte PrSOMS segmentée par la CAH.*

5 Conclusion

Notre approche présente une nouvelle approche pour capturer et modéliser l'information topographique présente dans les données séquentielles. L'approche proposée offre également un nouvel outil de visualisation topographique pour l'ensemble des données séquentielles. Elle est très bien adaptée pour les séquences multidimensionnelles (observations non i.i.d) et maintient un faible coût de calcul. L'approche proposée est évaluée en utilisant des données réelles issues de l'Institut Nationale de l'Audiovisuel (INA). Les résultats montrent une précision temporelle conforme à la vérité terrain et bien meilleure que celle indiquée dans les guides de programmes TV fournis par les chaînes de télévision. Comme perspectives, nous voulons appliquer notre approche sur des séquences binaires et catégorielles et exploiter le pouvoir de visualisation de notre approche.

Remerciement : Ce travail a été réalisé dans le cadre d'une thèse CIFRE avec l'INA. Nous remercions vivement Monsieur Jean-Hugues CHENOT pour ses remarques pertinentes.

Références

Anouar, F., F. Badran, et S. Thiria (1997). Self-organizing map, a probabilistic approach. In *Proceedings of WSOM'97-Workshop on Self-Organizing Maps, Espoo, Finland June 4-6*, pp. 339–344.

Bacciu, D., A. Micheli, et A. Sperduti (2010). Compositional generative mapping of structured data. In *Proceedings of International Joint Conference on Neural Networks*, IJCNN'10, pp. 1–8.

Bengio, Y. et P. Frasconi (1994). An input output hmm architecture. In *NIPS*, pp. 427–434.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA).

Bouchaffra, D. (2008). Embedding hmm's-based models in a euclidean space : The topological hidden markov models. In *ICPR08*, pp. 1–4.

Buzan, D., S. Sclaroff, et G. Kollios (2004). Extraction and clustering of motion trajectories in video. In *In International Conference on Pattern Recognition*, pp. 521–524.

Hagenbuchner, M., R. Sperduti, A. C. Tsoi, et S. Member (2003). A self-organizing map for adaptive processing of structured data. *IEEE Transactions on Neural Networks 14*, 491–505.

Kohonen, T. (2001). *Self-organizing Maps*. Springer Berlin.

Lebbah, M., N. Rogovschi, et Y. Bennani (2007). Besom : Bernoulli on self organizing map.

Luttrel, S. P. (1994). A bayesian analysis of self-organizing maps. *Neural Computing 6*, 767 – 794.

Olier, I. et A. Vellido (2008). Advances in clustering and visualization of time series using gtm through time. *Neural Netw. 21*, 904–913.

Prat, F., A. Marzal, S. MartÌn, R. Ramos-garijo, et M. J. C. Bleda (2009). A template-based recognition system for on-line handwritten characters. *Journal of Information Science and Engineering 25*, 779–791.

Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE 77*(2), 257–286.

Strickert, M. et B. Hammer (2003). Neural gas for sequences. In *Proceedings of the Workshop on Self-Organizing Networks (WSOM), Kyushu Institute of Technology*, pp. 53–57.

Verbeek, J., N. Vlassis, et B. Krose (2005). Self-organizing mixture models. *Neurocomputing 63*, 99–123.

Yamaguchi, N. (2010). Self-organizing hidden markov models. In K. Wong, B. Mendis, et A. Bouzerdoum (Eds.), *Neural Information Processing. Models and Applications*, Volume 6444 of *Lecture Notes in Computer Science*, pp. 454–461. Springer Berlin / Heidelberg.

Summary

We present a generative approach to learn a new probabilistic Self-Organizing Map (Pr-SOMS) for non independent and non identically distributed data sets. Our model defines a low dimensional manifold allowing friendly visualizations. To yield the topology preserving maps, our model has the SOM like learning behavior with the advantages of probabilistic models. This new paradigm uses HMM (Hidden Markov Models) formalism and introduces topological relationships between the states. This allows us to take advantage of all the known classical views associated to topographic map. We demonstrate our approach on the real-world data issued from French National Audiovisual Institute.