

A Generative Probabilistic Approach to Visualizing Sets of Symbolic Sequences

Peter Tiño
School of Computer Science
The University of Birmingham
Birmingham B15 2TT, UK
P.Tino@cs.bham.ac.uk

Ata Kabán
School of Computer Science
The University of Birmingham
Birmingham B15 2TT, UK
A.Kaban@cs.bham.ac.uk

Yi Sun
Faculty of Eng & Info Sciences
University of Hertfordshire
Hatfield, AL10 9AB, UK
Y.2.Sun@herts.ac.uk

ABSTRACT

There is a notable interest in extending probabilistic generative modeling principles to accommodate for more complex structured data types. In this paper we develop a generative probabilistic model for visualizing sets of discrete symbolic sequences. The model, a constrained mixture of discrete hidden Markov models, is a generalization of density-based visualization methods previously developed for static data sets. We illustrate our approach on sequences representing web-log data and chorals by J.S. Bach.

Categories and Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms: Algorithms, Design, Theory

Keywords: Hidden Markov model, latent space models, topographic mapping, EM algorithm

1. INTRODUCTION

Topographic visualisation techniques have been an important tool in multi-variate data analysis and data mining. Generative probabilistic approaches [2, 3, 7] have been developed and demonstrated to offer numerous advantages over non-probabilistic alternatives in terms of a flexible and technically sound framework that makes various extensions possible in a principled manner.

However, in their current form, most of these methods make an assumption that observation data can be represented in the form of a set of i.i.d. unstructured numerical vectors. While this may be a reasonable assumption in some cases, many of the most recent practical problems face us with the notion of structured observation types. This creates new challenges for data analysis research. Algorithms that are able to discover structure in sample sets of such structured entities need to be developed. Practical examples include various user profiling tasks, where, in the simplest case, one observation consists of a log trace left by a user as

a result of interacting with an electronic environment.

There has been a notable interest in extending probabilistic generative modeling principles to accommodate for more complex structured data types [4, 6, 13, 14]. The work in [4] essentially provides a method of visualisation of user navigation sequences based on clustering of Markov chains. Probabilistic clustering of hidden Markov models have also been developed [13] and applied to user navigation modelling in [14]. In addition to clustering models, the work in [6] proposes a computationally efficient convex distributed dynamic model for profiling and prediction of dynamic user activity.

However, none of these methods provide features for a topographic organization of these more complex data types. Topographic mappings require nonlinear distributed models in order to preserve the core of the information contained in the possibly complex and heterogeneous set of structured observations in a 2D visualisation plane.

The necessity of visualising non-vectorial structured types of data has been recognised in the literature and there are non-probabilistic, SOM-based approaches to e.g. visualising time series [5, 9]. In [5], a self organising map of first order Markov chains is developed and it is pointed out that the Euclidean distance measures initially employed in the non-probabilistic SOM method are not appropriate for clustering or visualisation based on probabilistic models. A Kullback-Leibler divergence is then employed within the SOM, however, this heuristic does not follow from a consistent model formulation.

In this paper we develop a consistent generative probabilistic model for visualizing sets of discrete symbolic sequences, where an appropriate divergence measure is defined by the noise model¹. The model is a constrained mixture of discrete hidden Markov models (HMM) [11]), where the constraint, introduced in the spirit of [2, 7], provides the model with topographic organisation capabilities.

The remainder of the paper is organised as follows: Section 2 introduces the model and the training algorithm. Section 3 provides experimental illustration of the method on two real world data sets: melodic lines of chorales by J.S. Bach and web navigation sequences. We conclude the paper with a brief summary of the key ideas and findings in section 4.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

KDD'04, August 22–25, 2004, Seattle, Washington, USA.
Copyright 2004 ACM 1-58113-888-1/04/0008 ...\$5.00.

¹Empirical Kullback-Leibler divergence between the unknown true distribution that generated the data item (sequence) and the reference hidden Markov model.

2. A LATENT TRAIT MODEL FOR DISCRETE HMMS

Consider a set of symbolic sequences over the alphabet of S symbols, $\mathcal{S} = \{1, 2, \dots, S\}$. The sequences can represent e.g. melodic lines, or traces of web users requested by a population of users. The n -th sequence will be denoted by $\mathbf{s}^{(n)} = (s_t^{(n)})_{t=1:T_n}$ where $n = 1 : N$ and T_n denotes the length of the n -th sequence. The lengths of the sequences may vary. Consider further an ($L=2$)-dimensional latent space $[-1, 1]^2$. The aim is to represent each sequence as using the latent space, such that the important overall characteristics of the set of sequences are revealed. A natural way of achieving this is to impose a maximum entropy (uniform) distribution over the latent space. For tractability reasons it is convenient to discretize the latent space into a regular grid of C points $\mathbf{x}_1, \dots, \mathbf{x}_C$. These sample (grid) points are analogous to the nodes of a Self Organising Map. With each grid point \mathbf{x}_c , we associate a generative distribution over sequences $p(\mathbf{s}|\mathbf{x}_c)$.

Assuming the sequences $\mathbf{s}^{(n)}$, $n = 1 : N$, were independently generated, the data likelihood of our model is

$$\mathcal{L} = \prod_{n=1}^N p(\mathbf{s}^{(n)}) = \prod_{n=1}^N \frac{1}{C} \sum_{c=1}^C p(\mathbf{s}^{(n)}|\mathbf{x}_c). \quad (1)$$

In order account for temporal dependency the sequences, we let the noise terms $p(\mathbf{s}|\mathbf{x}_c)$ to take the form of hidden Markov models with K hidden states [11]:

$$p(\mathbf{s}^{(n)}|\mathbf{x}_c) = \sum_{\mathbf{h}} p(h_1|\mathbf{x}_c) \prod_{t=2}^{T_n} p(h_t|h_{t-1}, \mathbf{x}_c) \prod_{t=1}^{T_n} p(s_t^{(n)}|h_t, \mathbf{x}_c), \quad (2)$$

where \mathbf{h} is the set of all T_n -tuples over the K hidden states.

The (logarithm of the) data likelihood needs now to be maximised. As there are hidden variables in the model, an EM-type solution will be adopted. According to the EM methodology, the expectation of the complete log likelihood needs to be maximised. The complete latent-centred conditional data distribution factorises into several multinomials:

$$\begin{aligned} p(\mathbf{s}^{(n)}, \mathbf{h}^{(n)}|\mathbf{x}_c) &= \prod_{k=1}^K p(h_1 = k|\mathbf{x}_c)^{\delta(h_1=k|\mathbf{s}^{(n)}, \mathbf{x}_c)} \\ &\prod_{t=2}^{T_n} \prod_{l=1}^K \prod_{k=1}^K p(h_t = k|h_{t-1} = l, \mathbf{x}_c)^{\delta(h_t=k, h_{t-1}=l|\mathbf{s}^{(n)}, \mathbf{x}_c)} \\ &\prod_{t=1}^{T_n} \prod_{k=1}^K p(s_t^{(n)}|h_t = k, \mathbf{x}_c)^{\delta(h_t=k|\mathbf{s}^{(n)}, \mathbf{x}_c)}. \end{aligned}$$

In order to have the HMM components topologically organised — e.g. on a two-dimensional equidistant grid — we will, in the spirit of [2, 7], constrain the mixture of HMMS,

$$p(\mathbf{s}) = \frac{1}{C} \sum_{c=1}^C p(\mathbf{s}|\mathbf{x}_c),$$

by requiring that the HMM parameters be generated through a parameterised *smooth* nonlinear mapping from the latent space into the HMM parameter space. In particular

$$\begin{aligned} \boldsymbol{\pi}_c &= \{p(h_1 = k|\mathbf{x}_c)\}_{k=1:K} \\ &= \{g_k(\mathbf{A}^{(\boldsymbol{\pi})} \boldsymbol{\phi}(\mathbf{x}_c))\}_{k=1:K} \end{aligned}$$

$$\begin{aligned} \mathbf{T}_c &= \{p(h_t = k|h_{t-1} = l, \mathbf{x}_c)\}_{k,l=1:K} \\ &= \{g_k(\mathbf{A}^{(\mathbf{T}_l)} \boldsymbol{\phi}(\mathbf{x}_c))\}_{k,l=1:K} \end{aligned}$$

$$\begin{aligned} \mathbf{B}_c &= \{p(s_t^{(n)} = s|h_t = k, \mathbf{x}_c)\}_{s=1:S, k=1:K} \\ &= \{g_s(\mathbf{A}^{(\mathbf{B}_k)} \boldsymbol{\phi}(\mathbf{x}_c))\}_{s=1:S, k=1:K} \end{aligned}$$

where

- the function $g(\cdot)$ is the softmax function, which is the canonical inverse link function of multinomial distributions and $g_k(\cdot)$ denotes the k -th component returned by the softmax, i.e.

$$g_k \left((a_1, a_2, \dots, a_q)^T \right) = \frac{e^{a_k}}{\sum_{i=1}^q e^{a_i}}, \quad k = 1, 2, \dots, q,$$

- $\mathbf{x}_c \in \mathcal{R}^2$ is the c -th grid point (representing the c -th sample from a uniformly distributed latent variable over the continuous visualisation space² $[-1, 1]^2$), $c = 1 : C$,
- $\boldsymbol{\phi}(\cdot) = (\phi_1(\cdot), \dots, \phi_M(\cdot))^T, \phi_m(\cdot) : \mathcal{R}^2 \rightarrow \mathcal{R}$ is an ordered set of M non-parametric nonlinear smooth basis functions (typically RBFs),
- the matrices $\mathbf{A}^{(\boldsymbol{\pi})} \in \mathcal{R}^{K \times M}$, $\mathbf{A}^{(\mathbf{T}_l)} \in \mathcal{R}^{K \times M}$ and $\mathbf{A}^{(\mathbf{B}_k)} \in \mathcal{R}^{S \times M}$ are free parameters of the model.

The expectation (with respect to the posterior distribution over the hidden variables, given the observed data) of the complete data log likelihood (relative likelihood [2]) is

$$\begin{aligned} Q &= \sum_{n=1}^N \sum_{c=1}^C p(\mathbf{x}_c|\mathbf{s}^{(n)}) \\ &\left[\sum_{k=1}^K p(h_1 = k|\mathbf{s}^{(n)}, \mathbf{x}_c) \log p(h_1 = k|\mathbf{x}_c) \right. \\ &+ \sum_{t=2}^{T_n} \sum_{l=1}^K \sum_{k=1}^K p(h_t = k, h_{t-1} = l|\mathbf{s}^{(n)}, \mathbf{x}_c) \\ &\quad \left. \log p(h_t = k|h_{t-1} = l, \mathbf{x}_c) \right. \\ &+ \left. \sum_{t=1}^{T_n} \sum_{k=1}^K p(h_t = k|\mathbf{s}^{(n)}, \mathbf{x}_c) \log p(s_t^{(n)}|h_t = k, \mathbf{x}_c) \right] \end{aligned}$$

Substituting the above quantities and solving stationary equations w.r.t. all parameters, we obtain the following algorithm:

2.1 Algorithm

- E step:
 - For each sequence $n = 1 : N$ and for each grid point \mathbf{x}_c , $c = 1 : C$, compute
 - * $\gamma_{nkt}^{(c)} = p(h_t = k|\mathbf{s}^{(n)}, \mathbf{x}_c)$, $t = 1 : T_n$
 - * $\omega_{nkl}^{(c)} = p(h_t = k, h_{t-1} = l|\mathbf{s}^{(n)}, \mathbf{x}_c)$, $t = 1 : T_n$.

Both quantities are determined by the forward-backward algorithm [11], using $\boldsymbol{\Theta}_c = \{\boldsymbol{\pi}_c, \mathbf{T}_c, \mathbf{B}_c\}$, parameters of the c -th HMM.

²uniform prior distribution of latent classes encourages full use of the available visualization space

- Compute ‘responsibilities’ of HMMs corresponding to grid points \mathbf{x}_c , $c = 1 : C$, for sequences $\mathbf{s}^{(n)}$, $n = 1 : N$,

$$r_{cn} = p(\mathbf{x}_c | \mathbf{s}^{(n)}) = \frac{p(\mathbf{s}^{(n)} | \mathbf{x}_c)}{\sum_{c'} p(\mathbf{s}^{(n)} | \mathbf{x}_{c'})},$$

where $p(\mathbf{s}^{(n)} | \mathbf{x}_c)$ is the likelihood of the n -th sequence under the c -th HMM.

- **M step:** Solve the following non-linear equations

$$\begin{aligned} \frac{\partial Q}{\partial \mathbf{A}_k^{(\boldsymbol{\pi})}} &= \sum_{n=1}^N \sum_{c=1}^C r_{cn} (\gamma_{nk1}^{(c)} - g_k(\mathbf{A}^{(\boldsymbol{\pi})} \boldsymbol{\phi}_c)) \boldsymbol{\phi}_c^T \\ &= 0 \\ \frac{\partial Q}{\partial \mathbf{A}_k^{(\mathbf{T}_l)}} &= \sum_{n=1}^N \sum_{c=1}^C r_{cn} \\ &\quad \left[\sum_{t=2}^{T_n} \omega_{nkt}^{(c)} - g_k(\mathbf{A}^{(\mathbf{T}_l)} \boldsymbol{\phi}_c) \sum_{t=2}^{T_n} \gamma_{n,l,t-1}^{(c)} \right] \boldsymbol{\phi}_c^T \\ &= 0, \quad l = 1 : K \\ \frac{\partial Q}{\partial \mathbf{A}_s^{(\mathbf{B}_k)}} &= \sum_{n=1}^N \sum_{c=1}^C r_{cn} \\ &\quad \left[\sum_{t=1 \wedge s_t^{(n)}=s}^{T_n} \gamma_{nkt}^{(c)} - g_s(\mathbf{A}^{(\mathbf{B}_k)} \boldsymbol{\phi}_c) \sum_{t=1}^{T_n} \gamma_{n,k,t}^{(c)} \right] \boldsymbol{\phi}_c^T \\ &= 0, \quad k = 1 : K, \end{aligned}$$

where $\mathbf{A}_i^{(\cdot)}$ denotes the i -th row of the parameter matrix $\mathbf{A}^{(\cdot)}$.

2.2 Visualizing symbolic sequences

Having trained the model on a set of sequences $\mathbf{s}^{(1)}$, $\mathbf{s}^{(2)}$, ..., $\mathbf{s}^{(N)}$, each sequence can now be represented by a point in the latent space – the mean of the posterior distribution over the latent space, given that sequence:

$$Proj(\mathbf{s}^{(n)}) = \sum_{c=1}^C \mathbf{x}_c p(\mathbf{x}_c | \mathbf{s}^{(n)}) = \sum_{c=1}^C \mathbf{x}_c r_{cn}.$$

This way, each sequence is mapped to one point in the 2D visualisation space. Because dynamic models have been employed as the noise models of the generative topographic mapping, dynamic structure of the sequences is the main feature that determines the notion of ‘closeness’ of sequence representations in the latent space.

3. EXPERIMENTS

In the experiments reported below the latent space centres \mathbf{x}_c were positioned on a regular 10×10 square grid ($C = 100$) and there were $M = 16$ basis functions ϕ_i . The basis functions were spherical Gaussian functions of the same width $\sigma = 1.0$. The basis functions were centred on a regular 4×4 square grid, reflecting uniform distribution of the latent classes. We account for a bias term by using an additional constant basis function $\phi_{17}(\mathbf{x}) = 1$.

Free parameters of the model were randomly initialized in the interval $[-1, 1]$. Training consisted of repeating EM integrations. Typically, the likelihood levelled up after 30-50 EM cycles.

3.1 Visualisation of Melodic Lines of Bach Chorals

In this experiment we visualize a set of 100 chorales by J.S. Bach [10]. We extracted the melodic lines – pitches are represented in the space of one octave, i.e. the observation symbol space consists of 12 different pitch values. Temporal structure of the sequences is the essential feature to be considered when organizing the data items in any sensible manner.

Figure 1 shows the posterior mean mapping obtained with our model. The method has essentially discovered the natural topography of the key signatures, corroborated with similarities of melodic motives. The upper right region contains the mapping of melodic lines that utilise keys with sharps. Coming towards the center region of the visualisation space we gradually find keys with less and less sharps to natural keys. There are no sharps or flats in the center and upper right regions of the plot, while the lower region of the plot is concerned with flats. The number of flats increases from left to right. The melodies can contain sharps and flats other than those included in their key signature due to both modulation and ornaments.

More interesting is to observe sub-groupings created according to melodic motives (patterns). The three melodic lines closest to the left boundary of the plot all contain a very characteristic and tense expressive melodic pattern **g-f#-bb-a** (that has interval of 4-), motive that can only be found in minor keys and is in general employed rather infrequently and with good musical reason only. Interestingly, the two closest points to those three do also contain an alleviated version of the same pattern (**g-f#-g-a-bb-a**), where the interval of 4- (**f#-bb**) is not explicit. Moreover, the closest point of the cluster from the upper region of the plot has (after a key modulation) a reversed form of this motive (**g-f#-e-d#-e**) that is far not as tense as the ones previously mentioned. The (this time descending) 4- (**g-d#**) is now ‘resolved’ (to **e**) within the motive.

To conclude, the benefit of the topographical representation is twofold: **(1)** More generally, it offers a means of organising an otherwise possibly tedious set of data structures in an intuitively understandable compact form and **(2)** for this specific example it also has the potential of providing a way of automatically addressing the phenomenon known as enharmony [1]. Enharmony refers to the situation when the same physical pitch value may have different musical interpretations as a function of the context (key and temporal structure). For example, a **c#** and a **db** are both encoded with the same number in a MIDI file, and are indeed physically the same frequency in a tempered intonation system. This is achieved by featuring the *context* in which the pitches appear. Correctly recognising enharmonics is essential e.g. in automatic chord annotation.

3.2 Visualisation of Web Navigation Sequences

The data considered in this experiment is a subset of the msnbc.com user navigation collection initially employed in [4] and also used in [6]. There are 1,480 browsing sessions totalling 119,667 page requests in this data set. The sessions consists of navigation patterns by users who visited at least 9 of the 17 page categories (frontpage, news, tech, local, opinion, on-air, misc, weather, msn-news, health, living, business, msn-sports, sports, summary, bbs, travel). From the analysis in [6] using Markov chains, where the state

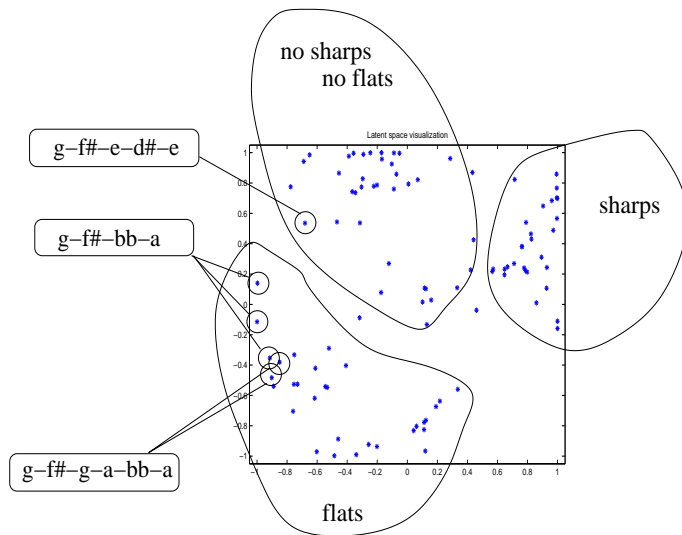


Figure 1: Visualization of melodic lines of 100 chorals by J.S. Bach.

space consisted of the above page categories, it transpires, that state repetition is a common feature of all browsing behaviours. Therefore, in this experiment, rather than concentrating on individual page categories, we consider a different alphabet. It will allow us to visually detect browsing behaviours in terms of the user’s ‘hunger for novelty’. To this end, we have defined three symbols: ‘1’: repeat the last category request ‘2’: return to category requested two moves before, ‘3’: all the other cases.

The browsing sessions are visualised in figure 2. To understand the plot better we also show the state transition probabilities and the emission probabilities of each of the $10 \times 10 = 100$ hidden Markov models underlying the visualization system. State transition structures are shown in figure 3(a) as a grid of maps of $K \times K = 2 \times 2$ state transition matrices $p(h_t = k | h_{t-1} = l, \mathbf{x}_c)$. Figure 3(b) presents emission probabilities in a grid of $S \times K = 3 \times 2$ matrices $p(s|k, \mathbf{x}_c)$.

Strong structure of emission probabilities is clearly visible in figure 3(b). The second hidden state is devoted almost exclusively to symbol 1 - ‘repeat the last category request’. Indeed, as mentioned earlier, category repetition is a common feature of all browsing behaviours (see [6]). In general, the first hidden state takes care of symbol 3 that encompasses all browsing behaviours different from ‘repeat the last category request’ and ‘return to category requested two moves before’. Such patterns can potentially include non-trivial navigation histories. Symbol 2 (‘return to category requested two moves before’) is less frequent than symbols 1 and 3 and is usually explained by the first state.

While the emission structure in figure 3(b) tells us about the marginal frequencies of symbols in sequences captured by the underlying HMMs, it is the state-transition structure in figure 3(a) that determines the temporal correlations between the symbols, i.e. the lengths of continuous blocks of symbols generated from one state. The stronger is the self-loop $p(h_t = k | h_{t-1} = k, \mathbf{x}_c)$ in state k of HMM c , the longer blocks of symbols favoured by that state (i.e. with higher emission probabilities $p(s|k, \mathbf{x}_c)$) are admissible. Transition probabilities close to $1/K = 0.5$ indicate a possi-

bility for symbol production arising from highly oscillating hidden states.

Moving from top to bottom of the latent space, we observe almost independent hidden states, with little chance of mutual transitions. Then the first state loses mass in its self-transition in favour of the second state, which becomes almost a trap state. Towards the bottom of the latent space, the first state recovers its power through a band of mixing patterns over the two hidden states. The mixing is strongest in the vicinity of the lower-left corner of the latent square.

Top of the plot is reserved for sequences containing long blocks of consecutive 1s and subsequences of 2s and 3s. Right part of the top cluster contains sequences with potentially long blocks of 1s and 2s. Sequences mapped in the left part of the top cluster contain potentially long blocks of 1s and 3s. Sequences corresponding to browsing within a single category are represented in the center of the plot. Lower-left corner is devoted to sequences of broad exploratory browsing behaviour, where all kinds of moves are possible, without longer persistent blocks of the same type of navigation behaviour. Dense cluster of sequences in the middle of the bottom part of the plot represents navigation patterns containing long periods of staying within the same category (consecutive 1s), interleaved with long periods of possibly non-trivial inter-topic search (consecutive 3s). Such sequences can also be found in the left part of the top cluster. In this sense, there is a hint of cylindrical organization of the latent space.

4. DISCUSSION

We have presented a generative probabilistic model for visualizing sets of discrete symbolic sequences. The model is essentially a constrained mixture of hidden Markov models that allows us to represent non-Markovian dynamical structures in a two-dimensional visualisation plane. Formally the model is a generalization of density-based visualization methods previously developed for static data sets [7]. We illustrated the model on sequences representing web-log data and chorals by J.S. Bach. We experimented with hidden Markov noise models with more than two hidden states,

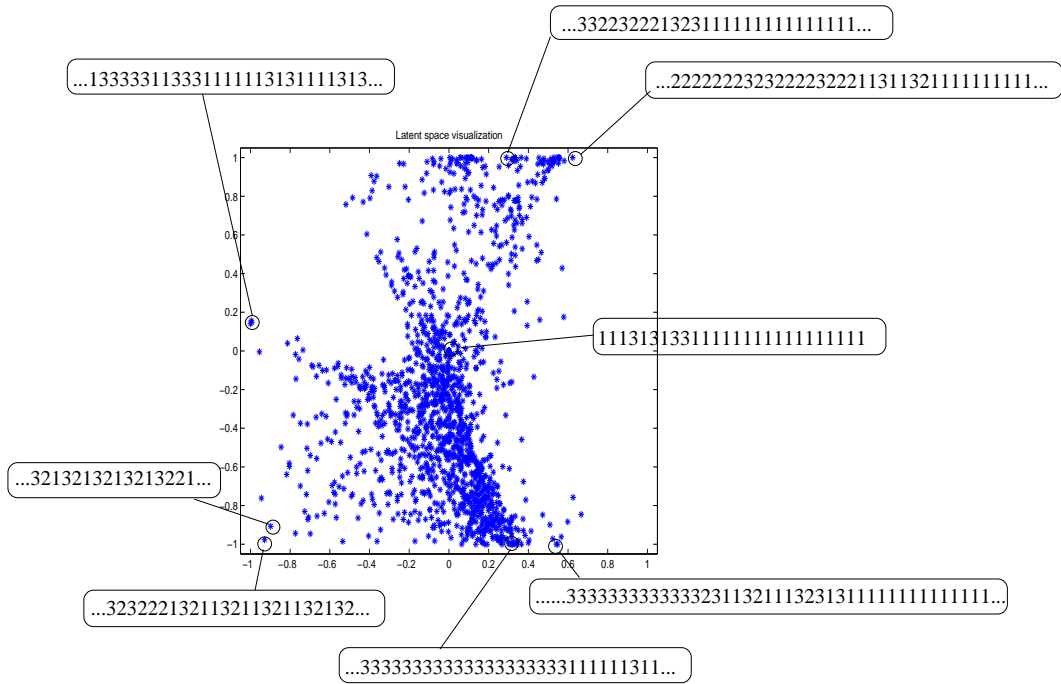


Figure 2: Visualization of web navigation sequences. For selected sequence representations we show a typical subsequence contained in the projected sequence.

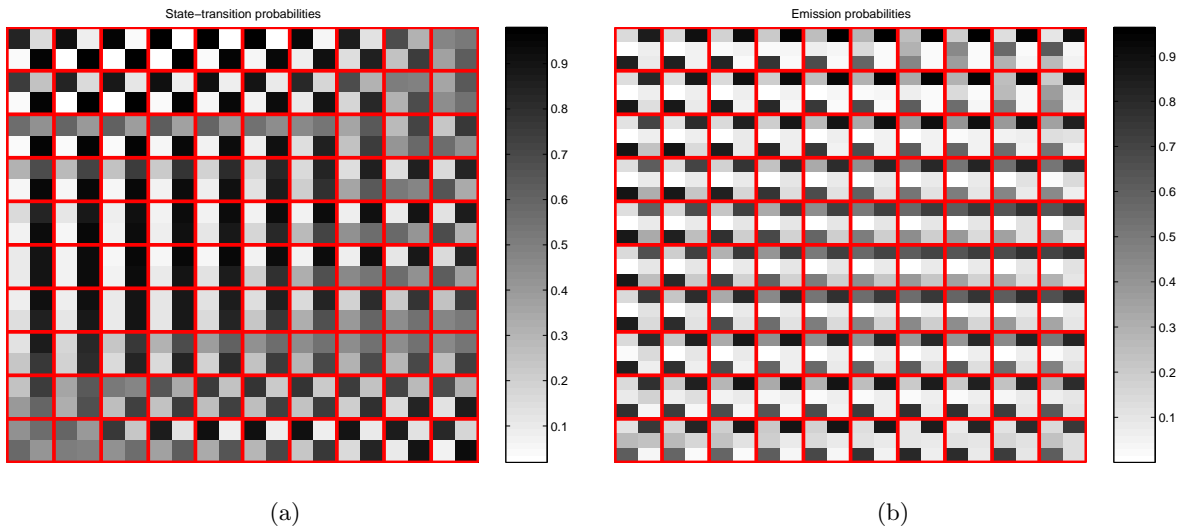


Figure 3: State transition (a) and emission (b) probabilities of HMMs underlying the visualization system in the web navigation experiment.

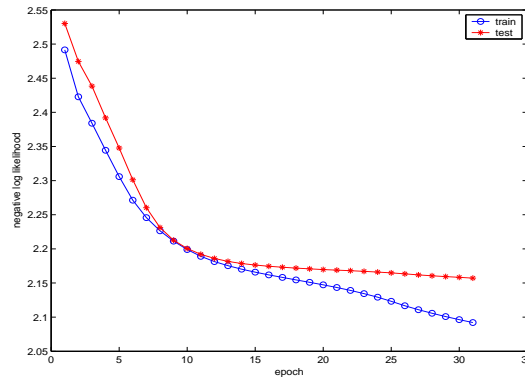


Figure 4: Evolution of negative log-likelihood per symbol measured on the training (o) and test (*) sets in the Bach chorals experiment.

however, the structure extracted by two-state models was already rich and interesting enough.

Our model constitutes a principled approach to visualization of sets of symbolic sequences. Probabilistic formulation enables the model to deal with e.g missing data, hierarchy building, or model selection in a consistent manner. Visualization plots can be naturally interpreted by plotting the state-transition and emission structures of the hidden Markov noise models corresponding to local regions of the latent space. In this sense, hidden Markov models are a much more viable option than standard fixed-order Markov noise models – going beyond the first-order Markov chain structure can prohibitively increase the number of states.

When the task we are facing is, for example, building a good probabilistic model for a given data set of sequences, without any concern for data visualization, then a suitable approach may be to use e.g. mixtures of HMMs, with appropriately chosen number of mixture components using a model selection technique. On the other hand, for *model based* visualization of sequential data, we may use many HMM components, but constrain them with a tight two-dimensional grid neighborhood structure. Such a constrained mixture of HMM may not be able to compete with appropriately constructed (probably smaller) unconstrained mixture of HMM on the grounds of density modeling, but it is suitable for data visualization and importantly, the tight grid topology prevents constrained models with many components (suitable for high-quality visualization) from excessively overfitting the data. The issue of data explanation vs. data prediction is covered e.g. in [12]. To evaluate tendency of our model to overfit the training data, we split the 100 Bach chorals into training and test sets containing 80 and 20 sequences, respectively. The model is trained solely on training sequences. Figure 4 shows the evolution of negative log-likelihood (NLL) per symbol measured on the training and test sets. After the 12th training epoch, the test set NLL stops decreasing, but crucially, due to the constrained nature of our model, it does not tend to increase at the expense of better modeling of the training data.

Scaling is, however, an issue. The E-step complexity is $O(NCTK^2)$. Obviously, a generative topographic formulation with fixed-order Markov chains would be much cheaper, since the noise models would not involve hidden variables. On the other hand, as argued above, the flexibility and interpretability of such formulations would be compromised.

Also, the model can be trained on a subsample of the available data and then used to visualize the whole data set. We are currently working on speeding up computations in our model through a variety of cheaper approximation techniques.

5. REFERENCES

- [1] J. Barnes, Bach’s Keyboard Temperament: Internal Evidence from the Well-Tempered Clavier, *Early Music*, 7(2), pp. 236-249, 1979.
- [2] C. Bishop, M. Svensén, and C. Williams, Generative Topographic Mapping, *Neural Computation*, 10(1), pp. 215-235, 1998.
- [3] C. Bishop, M. Svensén, and C. Williams, Developments of the Generative Topographic Mapping, *Neurocomputing*, 23, pp. 203-224, 1998.
- [4] I. Cadez, D. Heckerman, C. Meek, P. Smyth and White, S, Model-Based Clustering and Visualisation of Navigation Patterns on a Web Site, *Journal of Data Mining and Knowledge Discovery*, 7(4), pp.399-424 2003.
- [5] J. Hollmén, V. Tresp and O. Simula, A self-organizing map algorithm for clustering probabilistic models, Proc. of the Ninth International Conference on Artificial Neural Networks (ICANN’99), vol.2, pp. 946–951. IEE, 1999.
- [6] M. Girolami and A. Kabán. Simplicial Mixtures of Markov Chains: Distributed Modelling of Dynamic User Profiles. Advances in Neural Information Processing Systems (NIPS’03), in press.
- [7] A. Kabán and M. Girolami, A combined latent class and trait model for the analysis and visualization of discrete data, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8), pp. 859 -872, 2001.
- [8] A. Kabán and M. Girolami, A Dynamic Probabilistic Model to Visualise Topic Evolution in Text Streams. *Journal of Intelligent Information Systems*, 18(2–3), pp. 107–125, 2002.
- [9] J. Lampinen and E. Oja. Self-organising maps for spatial and temporal AR models. Proc. 6 SCIA, Scand. Conf. on Image Analysis, pp. 120–127, 1989.
- [10] Merz, C. and Murphy, P. UCI repository of Machine Learning Databases, 1998.
- [11] R.L. Rabiner and B.H. Juang. An introduction to hidden Markov models, *IEEE ASSP Magazine*, 3, pp. 4–16, 1986.
- [12] B. Ripley, Statistical Theories of Model Fitting. In Ch. Bishop, editor, *Neural Networks and Machine Learning (NATO ASI series III, Computer and System Sciences)*, pp 3–26. Springer Verlag, 1998.
- [13] P. Smyth, Mixtures of Hidden Markov Models, Advances in Neural Information Processing 9, M. C. Mozer, M. I. Jordan and T. Petsche (eds) Cambridge, MA: MIT Press, pp. 648–654, 1997.
- [14] A. Ypma and T. Heskes, Categorisation of web pages and user clustering with mixtures of hidden markov models. *WEBKDD* (pp. 31–43), 2002.