Factored Conditional Restricted Boltzmann Machines for Modeling Motion Style

Graham W. Taylor GWTAYLOR@CS.TORONTO.EDU
Geoffrey E. Hinton HINTON@CS.TORONTO.EDU
Department of Computer Science, University of Toronto, Toronto, Ontario M5S 2G4, Canada

Abstract

The Conditional Restricted Boltzmann Machine (CRBM) is a recently proposed model for time series that has a rich, distributed hidden state and permits simple, exact inference. We present a new model, based on the CRBM that preserves its most important computational properties and includes multiplicative three-way interactions that allow the effective interaction weight between two units to be modulated by the dynamic state of a third unit. We factor the three-way weight tensor implied by the multiplicative model, reducing the number of parameters from $O(N^3)$ to $O(N^2)$. The result is an efficient, compact model whose effectiveness we demonstrate by modeling human motion. Like the CRBM, our model can capture diverse styles of motion with a single set of parameters, and the three-way interactions greatly improve the model's ability to blend motion styles or to transition smoothly among them.

1. Introduction

Directed graphical models (or Bayes nets) have been a dominant paradigm in models of static data. Their temporal counterparts, Dynamic Bayes nets, generalize many existing models such as the Hidden Markov Model (HMM) and its various extensions. In all but the simplest directed models, inference is made difficult due to a phenomenon known as "explaining away" where observing a child node renders its parents dependent. An alternative to approximate inference in directed models is to use a special type of *undirected* model, the Restricted Boltzmann Machine (RBM) (Smolensky, 1986), that allows efficient, exact inference. The Restricted Boltzmann Machine has an efficient, approximate learning algorithm called contrastive divergence (CD) (Hinton, 2002). RBMs have been used in a variety of applications (Hinton & Salakhutdinov, 2006; Salakhutdinov et al., 2007) and their properties have become better understood over the last few years (Welling et al., 2005; Carreira-Perpinan & Hinton, 2005; Salakhutdinov & Murray, 2008). The CD learning procedure has also been improved (Tieleman, 2008).

A major motivation for the use of RBMs is that they can be used as the building blocks of deep belief networks (DBN), which are learned efficiently by training greedily, layer-by-layer. DBNs have been shown to learn very good generative models of handwritten digits (Hinton et al., 2006), but they fail to model patches of natural images. This is because RBMs have difficulty in capturing the smoothness constraint in natural images: a single pixel can usually be predicted very accurately by simply interpolating its neighbours. Osindero and Hinton (2008) introduced the Semi-restricted Boltzmann Machine (SRBM) to address this concern. The constraints on the connectivity of the RBM are relaxed to allow lateral connections between the *visible* units in order to model the pair-wise correlations between inputs, thus allowing the hidden units to focus on modeling higher-order structure. SRBMs also permit deep networks. Each time a new level is added, the previous top layer of units is given lateral conections, so, after the layer-by-layer learning is complete, all layers except the topmost contain lateral connections between units. SRBMs make it possible to learn deep belief nets that model image patches much better, but they still have strong limitations that can be seen by considering the overall generative model. The equilibrium sample generated at each layer influences the layer below by controlling its effective biases. The model would be much more powerful if the equilibrium sample at the higher level could control the lateral interactions at the layer below using a three-way, multiplicative relationship. Memisevic and Hinton (2007) introduced the gated CRBM, which permitted such multiplicative interactions and was able to learn rich, distributed representations of image transformations.

Appearing in *Proceedings of the 26^{th} International Conference on Machine Learning*, Montreal, Canada, 2009.

In this paper, we explore the idea of multiplicative interactions in a different type of CRBM (Taylor et al., 2007). Instead of gating lateral interactions with hidden units, we allow a set of context variables to gate the three types of connections ("sub-models") in the CRBM shown in Fig. 1. Our modification of the CRBM architecture does not change the desirable properties related to inference and learning but makes the model context-sensitive.

While our model is applicable to general time series where conditional data is available (e.g. seasonal variables for modeling rainfall occurrences, economic indicators for modeling financial instruments) we apply our work to capturing aspects of style in data captured from human motion (mocap). Taylor et al. (2007) showed that a CRBM could capture many different styles with a single set of parameters. Generation of different styles was purely based on initialization, and the model architecture did not allow control of transitions among styles nor did it permit style blending. By using style variables to gate the connections of a CRBM, we obtain a much more powerful generative model that permits controlled transitioning and blending. We demonstrate that in a conditional model, gating is superior to simply using labels to bias the hidden units, which is the technique most commonly applied to static models.

This paper is also part of a large body of work related to the separation of style and content in motion. The ability to separately specify the style (e.g. sad) and the content (e.g. walk to location A) is highly desirable for animators. Previous work has looked at applying user-specified style to an existing motion sequence (Hsu et al., 2005; Torresani et al., 2007). The drawback to these approaches is that the user must provide the content. We propose a generative model for content that adapts to stylistic controls. Recently, models based on the Gaussian Process Latent Variable Model (Lawrence, 2004) have been successfully applied to separate content and style in human motion (Wang et al., 2007). The advantage of our approach over such methods is that our model does not need to retain the training dataset (just a few frames for initialization) and is thus suitable for low-memory devices. Furthermore, training is linear in the number of frames, and so our model can scale up to massive datasets, unlike the kernel-based methods which are cubic in the number of frames. The rich, distributed hidden state of our model means that it does not suffer from the from the limited representational power of HMM-based methods (e.g. Brand & Hertzmann, 2000).

2. Background

2.1. Conditional RBMs

The CRBM (Fig. 1) is a non-linear generative model for time series data that uses an undirected model with binary

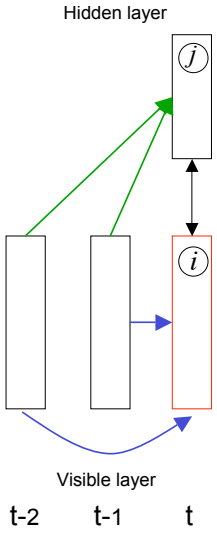


Figure 1. Architecture of the CRBM

latent variables, $\mathbf{h}$, connected to a collection of visible variables, $\mathbf{v}$. The visible variables can use any distribution in the exponential family (Welling et al., 2005), but for mocap data, we use real-valued Gaussian units (Freund & Haussler, 1992). At each time step t, $\mathbf{v}$ and $\mathbf{h}$ receive directed connections from the visible variables at the last N time-steps. To simplify the presentation, we will assume the data at $t-1,\ldots,t-N$ is concatenated into a "history" vector which we call $\mathbf{v_{<t}}$. We will use k to index the elements of $\mathbf{v_{<t}}$. The model defines a joint probability distribution over $\mathbf{v}_t$ and $\mathbf{h}_t$, conditional on $\mathbf{v}_{<t}$ and model parameters, θ:

$$p(\mathbf{v}_t,\mathbf{h}_t|\mathbf{v}_{<t},\theta)=\exp\left(-E\left(\mathbf{v}_t,\mathbf{h}_t|\mathbf{v}_{<t},\theta\right)\right)/Z$$

$$E=\sum_i\frac{(\hat{a}_{i,t}-v_{i,t})^2}{2\sigma_i^2}-\sum_j\hat{b}_{j,t}h_{j,t}-\sum_{ij}W_{ij}\frac{v_{i,t}}{\sigma_i}h_{j,t} \quad (1)$$

where Z is a constant called the partition function which is exponentially expensive to compute exactly. The dynamic biases, $\hat{a}_{i,t}=a_i+\sum_k A_{ki}v_{k,<t}$ and $\hat{b}_{j,t}=b_j+\sum_k B_{kj}v_{k,<t}$, express the net input from the past to the visible and hidden units, respectively. As is commonly done, we set $\sigma_i=1$.

Such an architecture makes on-line inference efficient and allows us to train by minimizing contrastive divergence (for details, see Hinton, 2002). Taylor et al. (2007) applied the CRBM to synthesize novel motion and perform on-line filling in of data lost during motion capture.

An important feature of the CRBM is that once it is trained, we can add layers like in a Deep Belief Network (Hinton et al., 2006). The previous layer CRBM is kept, and the sequence of hidden state vectors, while driven by the data, is treated as a new kind of "fully observed" data. The next

level CRBM has the same architecture as the first (though it has binary visible units and we can change the number of hidden units) and is trained in the exact same way. Upper levels of the network can then model more interesting higher-order structure. More layers aid in capturing multiple styles of motion, and permitting transitions among these styles (see Sec. 4).

2.2. Gated Conditional Restricted Boltzmann Machines

Memisevic and Hinton (2007) introduced a way of implementing multiplicative interactions in a conditional model. The gated CRBM was developed in the context of learning transformations between image pairs. The idea is to model an observation (the output) given its previous instance (the input) (e.g. neighbouring frames of video). The gated CRBM has two equivalent views: first, as gated regression (Fig. 2a), where hidden units can blend "slices" of a transformation matrix into a linear regression, and second as modulated filters (Fig. 2b) where input units gate a set of basis functions used to reconstruct the output. In the latter view, each setting of the input units defines an RBM (which means that conditional on the input, inference and learning in a gated CRBM are tractable). For ease of presentation,

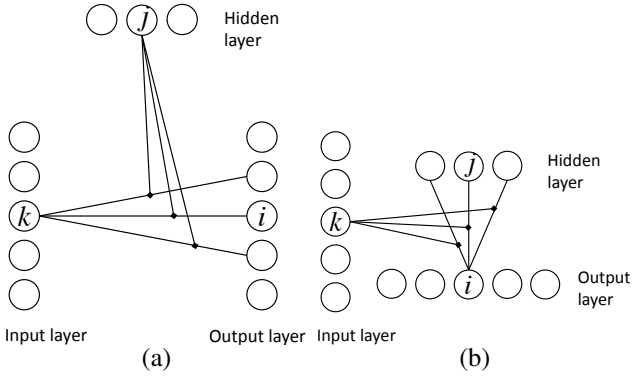


Figure 2. Two views of the Gated Boltzmann Machine. Reproduced from (Memisevic & Hinton, 2007).

let us consider the case where all input, output, and hidden variables are binary (the extension to real-valued input and output variables is straightforward). As in Eq. 1, the gated CRBM describes a joint probability distribution through exponentiating and renormalizing an energy function. This energy function captures all possible correlations between the components of the input, $\mathbf{x}$, the output, $\mathbf{v}$, and the hidden variables, $\mathbf{h}$:

$$E(\mathbf{v}, \mathbf{h}|\mathbf{x}, \theta) = -\sum_{ijk} W_{ijk} v_i h_j x_k - \sum_{ij} c_{ij} v_i h_j - \sum_i a_i v_i - \sum_j b_j h_j \quad (2)$$

where a_i, b_j index the standard biases on each unit and c_{ij} index the gated biases, which shift a unit conditionally. The parameters W_{ijk} are the components of a three-way weight tensor. The CD weight updates for learning a gated CRBM are similar to a standard RBM.

2.3. Factoring

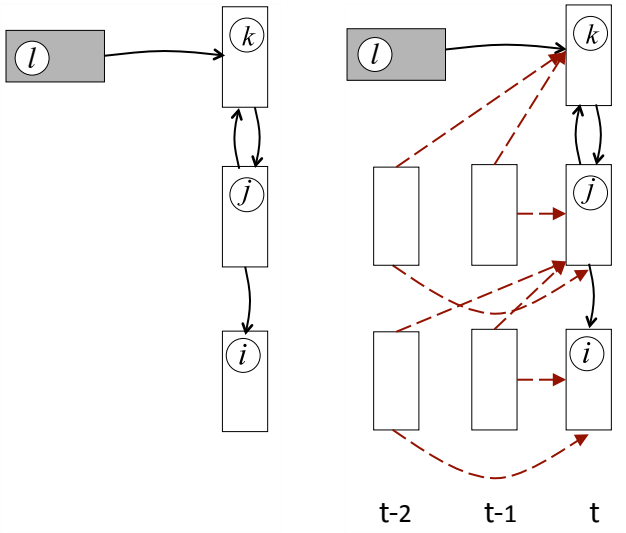
To model time series, we can consider the output of a gated CRBM to be the current frame of data, $\mathbf{v} = \mathbf{v}_t$, and the input to be the previous frame (or frames), $\mathbf{x} = \mathbf{v}_{<t}$. This means that the gated CRBM is a kind of autoregressive model where a transformation is composed from a set of simpler transformations. The number of possible compositions is exponential in the number of hidden units, but the componential nature of the hidden units prevents the number of parameters in the model from becoming exponential, as it would in a mixture model. Because of the three-way weight tensor, the number of parameters is cubic (assuming that the numbers of input, output and hidden units are comparable).

In many applications, including mocap, strong underlying regularities in the data suggest that structure can be captured using three-way, multiplicative interactions but with less than the cubically many parameters implied by the weight tensor. This motivates us to factor the interaction tensor into a product of pairwise interactions. If we apply the factoring to Eq. 2, the first term becomes $\sum_f \sum_{ijk} W^{\mathbf{v}}_{if} W^{\mathbf{h}}_{jf} W^{\mathbf{x}}_{kf} v_i h_j x_k$, where f indexes a set of deterministic factors. Superscripts differentiate the three types of pairwise interactions: $W^{\mathbf{v}}_{if}$ connect output units to factors (undirected), $W^{\mathbf{h}}_{jf}$ connect hidden units to factors (undirected), and $W^{\mathbf{x}}_{kf}$ connect input units to factors (directed). If the number of factors is comparable to the number of other units, this reduces the number of parameters from $O(N^3)$ to $O(N^2)$. Although factoring has been motivated by the introduction of multiplicative interactions, models that only involve pairwise interaction can also be factored.

3. A Style-Gated, Factored Model

We now consider modeling multiple styles of human motion using factored, multiplicative, three-way interactions. Hinton et al. (2006) showed that a good generative model of handwritten digits could be built by connecting a softmax label unit to the topmost hidden layer of a DBN (Fig. 3a). Clamping a label changed the energy landscape of the autoassociative model formed by the top two layers, such that performing alternating Gibbs sampling would produce a joint sample compatible with a particular digit class. It is easy to extend this modification to the CRBM, where discrete style labels bias the hidden units. In a CRBM, however, the hidden units also condition on information from

the past that is much stronger than the information coming from the label (Fig. 3b). The model has learned to respect consistency of styles between frames and so will resist a transition introduced by changing the label units.



(a) (b)

Figure 3. a) In a deep belief network, clamping the label units changes the energy function. b) In a conditional model, label information is swamped by the signal coming from the past.

As in the gated CRBM, we are motivated to let style change the *interactions* of the units as opposed to simply their effective biases. Memisevic (2008) used factored three-way interactions to allow the hidden units of a gated CRBM to control the effect of one video frame on the subsequent video frame. Figure 4 shows a different way of using factored three-way interactions to allow real-valued style features, derived from discrete style labels, to control three different sets of pairwise interactions. Like the standard CRBM (Eq. 1), the model defines a joint probability distribution over $\mathbf{v}_t$ and $\mathbf{h}_t$, conditional on the past N observations, $\mathbf{v}_{<t}$, and model parameters, θ. However, the distribution is also conditional on the style labels, $\mathbf{y}_t$. Similar to our discussion of the CRBM, we assume binary stochastic hidden units and real-valued visible units with additive, Gaussian noise. For notational ease, we assume $\sigma_i = 1$. The energy function is:

$$E\left(\mathbf{v}_t, \mathbf{h}_t | \mathbf{v}_{<t}, \mathbf{y}_t, \theta\right) = \frac{1}{2}\sum_i \left(\hat{a}_{i,t} - v_{i,t}\right)^2 - \sum_f \sum_{ijl} W^{\mathbf{v}}_{if} W^{\mathbf{h}}_{jf} W^{\mathbf{z}}_{lf} v_{i,t} h_{j,t} z_{l,t} - \sum_j \hat{b}_{j,t} h_{j,t}. \quad (3)$$

The three terms in Eq. 3 correspond to the three sub-models (coloured blue, red, and green, respectively in Fig. 4). For each sub-model, what was a matrix of weights is now replaced by three sets of weights connecting units to factors. The types of weights are differentiated again by superscripts. For example, the matrix of undirected weights in the standard CRBM, W_{ij}, has been replaced by three matrices involved in a factored, multiplicative interaction: $W^{\mathbf{v}}_{if}$, $W^{\mathbf{h}}_{jf}$, and $W^{\mathbf{z}}_{lf}$. The same process is applied to the other two sub-models. Note that the three sub-models may have a different number of factors (which we index by f, m, and n).

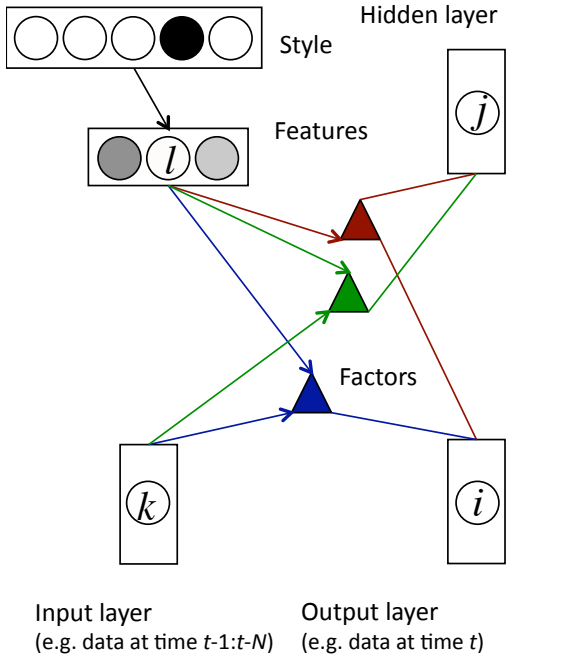


Figure 4. A factored CRBM whose interactions are gated by real-valued stylistic features.

The dynamic biases become:

$$\hat{a}_{i,t} = a_i + \sum_m A^{\mathbf{v}}_{im} \sum_k A^{\mathbf{v}_{<t}}_{km} v_{k,<t} \sum_l A^{\mathbf{z}}_{lm} z_{l,t}, \quad (4)$$

$$\hat{b}_{j,t} = b_j + \sum_n B^{\mathbf{h}}_{jn} \sum_k B^{\mathbf{v}_{<t}}_{kn} v_{k,<t} \sum_l B^{\mathbf{z}}_{ln} z_{l,t} \quad (5)$$

where the dynamic component of Eq. 4 and Eq. 5 is simply the total input to the visible/hidden unit via the factors. The total input is a three-way product between the input to the factors (coming from the past and from the style features) and the weight from the factors to the visible/hidden unit. The dynamic biases include a static component, $\mathbf{a}$ and $\mathbf{b}$. As in the gated CRBM, we could also add three types of gated biases, corresponding to the pairwise interactions in each of the sub-models. In our experiments, we have not used any gated biases.

The deterministic features, $\mathbf{z}_t$, are a linear function of the

"one-hot" encoded style labels, $\mathbf{y}_t$:

$$z_{l,t} = \sum_p R_{pl} y_{p,t}. \tag{6}$$

As with other models based on RBMs, the existence of the partition function means that maximum likelihood learning is intractable. Nonetheless, it is easy to compute a good approximation to the gradient of an alternative objective function called the contrastive divergence, which leads to a set of very simple gradient update rules. The updates for all W, A, and B parameters take the form:

$$\Delta X_{qr} \propto \sum_t \left(\langle \alpha_{q,t} \beta_{r,t} \gamma_{r,t} \rangle_0 - \langle \alpha_{q,t} \beta_{r,t} \gamma_{r,t} \rangle_K \right) \tag{7}$$

where $\alpha_{q,t}; q \in i, j, k, l$ is the unit connected to factor $r; r \in f, m, n$ by weight X_{qr}. Terms $\beta_{r,t}$ and $\gamma_{r,t}$ correspond to the total input that arrives at factor r from the two other types of units involved in the three-way relationship. $\langle \rangle_0$ is an expectation with respect to the data distribution, and $\langle \rangle_K$ is an expectation with respect to the joint distribution obtained from starting with a training vector clamped to the visibles and performing K steps of alternating Gibbs sampling (i.e. CD-K). Consider two concrete examples:

$$\Delta W^{\mathbf{v}}_{if} \propto \sum_t \left(\langle v_{i,t} \sum_j W^{\mathbf{h}}_{jf} h_{j,t} \sum_l W^{\mathbf{z}}_{lf} z_{l,t} \rangle_0 - \langle v_{i,t} \sum_j W^{\mathbf{h}}_{jf} h_{j,t} \sum_l W^{\mathbf{z}}_{lf} z_{l,t} \rangle_K \right), \tag{8}$$

$$\Delta A^{\mathbf{z}}_{lm} \propto \sum_t \left(\langle z_{l,t} \sum_i A^{\mathbf{v}}_{im} v_{i,t} \sum_k A^{\mathbf{v}_{<t}}_{km} v_{k,<t} \rangle_0 - \langle z_{l,t} \sum_i A^{\mathbf{v}}_{im} v_{i,t} \sum_k A^{\mathbf{v}_{<t}}_{km} v_{k,<t} \rangle_K \right). \tag{9}$$

The weights connecting labels to features, R, can simply be learned by backpropagating the gradients obtained by CD. Since these weights affect all three sub-models, their updates are more complicated. Applying the chain rule:

$$\begin{aligned} \Delta R_{pl} &\propto \sum_t \left(\langle C_{l,t} y_{p,t} \rangle_0 - \langle C_{l,t} y_{p,t} \rangle_K \right), \\ C_{l,t} &= \sum_f W^{\mathbf{z}}_{lf} \sum_i W^{\mathbf{v}}_{if} v_{i,t} \sum_j W^{\mathbf{h}}_{jf} h_{j,t} \\ &+ \sum_m A^{\mathbf{z}}_{lm} \sum_i A^{\mathbf{v}}_{im} v_{i,t} \sum_k A^{\mathbf{v}_{<t}}_{km} v_{k,<t} \\ &+ \sum_n B^{\mathbf{z}}_{ln} \sum_j B^{\mathbf{h}}_{jn} h_{j,t} \sum_k B^{\mathbf{v}_{<t}}_{kn} v_{k,<t}. \end{aligned} \tag{10}$$

The updates for the hidden and visible biases are the same as in the standard CRBM (Taylor et al., 2007).

3.1. Parameter sharing

In addition to the massive reduction in the number of free parameters obtained by factoring, further savings may be obtained by tying some sets of parameters together. In the fully parameterized model (Fig. 5a), there are 9 different sets (matrices) of weights but if we restrict the number of factors to be the same for each of the three sub-models, four sets of parameters are identical in dimension: the weights that originate from the inputs (past visible units), the outputs (visible units), the hidden units and the features. Any combination of the compatible parameters may be tied. Fig. 5b shows a fully-shared parameterization. This has slightly less than half the number of parameters of the fully parameterized model, assuming that the number of input, output, hidden, and feature units are comparable.

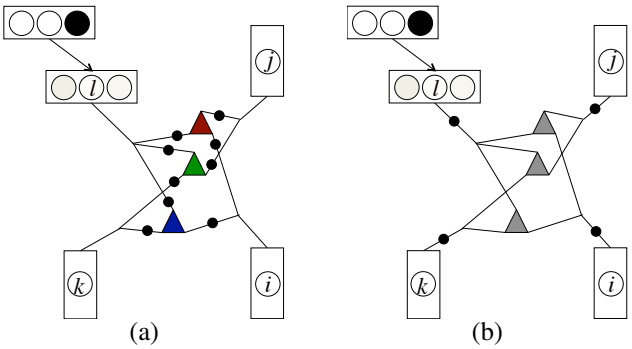


Figure 5. a) Fully parameterized model with each dot representing a different set of parameters and different colors denoting a different number of factors in each sub-model. b) Full parameter sharing where each dot represents a tied group of parameters.

In comparing different reduced parameterizations, tying only the feature-factor parameters, $W^{\mathbf{z}}_{lf}$, $A^{\mathbf{z}}_{lm}$, and $B^{\mathbf{z}}_{ln}$ led to models that synthesized the highest quality motion. When sharing the autoregressive weights, $A^{\mathbf{v}_{<t}}_{km}$ and $A^{\mathbf{v}}_{im}$, with non-autoregressive weights, $B^{\mathbf{v}_{<t}}_{km}$ and $W^{\mathbf{v}}_{if}$, respectively, we found that the component of the gradient related to the autoregressive model tended to dominate the weight update early in learning. This was due to the strength of the correlation between past and present compared to hidden and present or hidden and past. Witholding the autoregressive component of the gradient for the first 100 epochs, until the hidden units were able to extract interesting structure from the data, solved this problem. In our reported experiments we trained models with only the feature-factor parameters tied.

4. Experiments

Sections 4.1-4.2 report the results of training several models with data retrieved from the CMU Graphics Lab Motion Capture Database. We extracted a series of 10 stylized

walk sequences performed by subject 137. The walks were labeled as *cat, chicken, dinosaur, drunk, gangly, graceful, normal, old-man, sexy* and *strong*. We balanced the dataset by repeating the sequences 3-6 times (depending on the original length) so that our final dataset contained approximately 3000 frames of each style at 60fps. All rotations were converted to an exponential map representation. As in (Taylor et al., 2007), the root segment was expressed in a body-centred coordinate system which is invariant to ground-plane translations and rotations about the gravitational vertical. All data was scaled to have zero mean and unit variance. We refer the reader to videos of our synthesized data at `http://www.cs.toronto.edu/~gwtaylor/publications/icml2009/`.

4.1. Baseline: the CRBM

As a baseline, we trained two CRBM models following (Taylor et al., 2007) with the following exceptions:

- At each iteration of CD learning, we performed 10 steps of alternating Gibbs sampling (CD-10).
- We added a sparsity term to the energy function to gently encourage the hidden units, while driven by the data, to have an average activation of 0.2. This is the same kind of sparsity used in (Lee et al., 2008).
- At each iteration of CD learning, we added Gaussian noise with $\sigma = 1$ to each dimension of $\mathbf{v}_{<t}$.

All parameters used a learning rate of 10^{-3}, except for the autoregressive weights which used a learning rate of 10^{-5}.

4.1.1. 1-LAYER MODEL

A single-layer CRBM with 1200 hidden units and $N = 12$ was trained on the 10-style data for 200 epochs with the parameters being updated after every 100 training cases. Each training case was a window of 13 consecutive frames and the order of the training cases was randomly permuted. In addition to the real-valued mocap data, the hidden units received input from a "one-hot" encoding of the matching style label. Respecting the conditional nature of our application (generation of stylized motion, as opposed to, say classification) this label was not reconstructed during learning. After training the model, we generated motion by initializing with 12 frames of training data and holding the label units clamped to the style matching the initialization.

With a single layer we can generate high-quality motion of 9/10 styles (see the supplemental videos), however, the model fails to produce good generation of the *old-man* style. We believe that this relates to the subtle nature of this particular motion. In examining the activity of the hidden units over time while clamped to training data, we observed that the model devotes most of its hidden capacity to capturing the more "active" styles as it pays a higher cost for failing to model more pronounced frame-to-frame changes.

4.1.2. 2-LAYER MODEL

We also learned a deeper network by first training a CRBM with 600 binary hidden units and real-valued visible units and then training a second level CRBM with 600 binary hidden and 600 binary visible units. The data for training the second level CRBM was the activations of the hidden units of the first level CRBM while driven by the training data. We added style labels to the top layer while training the second level CRBM. The first model was trained for 300 epochs, and the second level was trained for 120 epochs. After training, the 2-hidden-layer network was able to generate high-quality walks of all styles, including *old-man* (see the supplemental videos). The second level CRBM layer effectively replaces the prior over the first layer of hidden units, $p(\mathbf{h}_t|\mathbf{v}_{<t}, \theta)$, that is implicitly defined by the parameters of the first CRBM. This provides a better model of the subtle correlations between the features that the first level CRBM extracts from the motion.

4.2. Modeling with Discrete Style Labels

Using the same 10-styles dataset, we trained a factored CRBM with Gaussian visible units whose parameters were gated by 100 real-valued features driven by the discrete style labels (Fig. 4). This model had 600 hidden units, 200 factors per sub-model and $N = 12$. Feature-to-factor parameters were also tied between sub-models. All parameters used a learning rate of 10^{-2}, except for the autoregressive parameters $A^{\mathbf{v}}_{im}$, $A^{\mathbf{v}_{<t}}_{km}$, $A^{\mathbf{z}}_{lm}$ and the label-to-feature parameters, R_{pl}, which used a learning rate of 10^{-3}. We trained the model for 500 epochs. After training the model, we tested its ability to synthesize realistic motion by initializing with 12 frames of training data and holding the label units clamped to the matching style. The single-layer model was able to generate stylized content as well as the 2-layer standard CRBM (see the supplemental videos). In addition, we were able to induce transitions between two or more styles by linearly blending the discrete style label from one setting to another over 200 frames[1]. We were further able to blend together styles (like *sexy* and *strong*) by applying a linear interpolation of the discrete labels. The resulting motion was more natural when a single style was dominant (e.g. an 0.8/0.2 blend). We believe this is simply a case of better performance when the desired motion more closely resembles the cases present in the training data set, so training on a few examples of blends should greatly improve their generation.

[1]The number of frames was selected empirically and provided a smooth transition, but the model is not sensitive to this number. A quick (e.g. frame-to-frame) change of labels will simply produce a "jerky" transition.

4.3. Modeling with Real-valued Style Parameters

The motions considered thus far have been described by a single, discrete label such as *gangly* or *drunk*. Motion style, however, can be characterized by multiple discrete labels or even continuous factors such as the level of flow, weight, time and space formally defined in Laban Movement Analysis (Torresani et al., 2007). In the case of multiple discrete labels, our real-valued feature units, $\mathbf{z}$, can receive input from multiple categories of labels. For continuous factors of style, we can connect real-valued style units to the real-valued feature units, or we can simply gate the model directly by the continous description of style.

To test this hypothesis, we trained a model exactly as in Sec. 4.2, but instead of gating connections with 100 real-valued feature units, we gated with 2 real-valued style descriptors that were conditioned upon at every frame. Again we trained with walking data, but the data was captured specifically for this experiment. One style unit represented the speed of walking and the other, the stride length. The training data consisted of nine sequences at 60fps, each approximately 6000 frames corresponding to the cross-product of (*slow*, *normal*, *fast*) speed and (*short*, *normal*, *long*) stride length. The corresponding labels each had values of 1, 2 or 3. These values were chosen to avoid the special case of all gating units being set at zero and nullifying the effective weights of the model.

After training for 500 epochs, the model could, as before, generate realistic motion according to the nine discrete combinations of speed and stride-length with which it was trained based on initialization and setting the label units to match the labels in the training set. Furthermore, the model supported both interpolation and extrapolation along the speed and stride length axes and did not appear overly sensitive to initialization (see the supplemental videos).

4.4. Quantitative Evaluation

In our experiments so far, we have sought a qualitative comparison to the CRBM, based on the realism of synthesized motion. We have also focused on the ability of a factored model with multiplicative interactions to synthesize transitions as well as interpolate and extrapolate among styles present in the training data set. The application does not naturally present a quantitative comparison, but in the past, other time series models have been compared by their performance on the prediction of either full or partial held-out frames (Taylor et al., 2007). We use the dataset first proposed by (Hsu et al., 2005) which consists of labeled sequences of seven types of walking: (*crouch*, *jog*, *limp*, *normal*, *side-right*, *sway*, *waddle*) each at three different speeds (*slow*, *medium*, *fast*). We preprocessed the data to remove missing or extremely noisy sections, and smoothed with a low-pass filter before downsampling from 120 to 30fps.

For each architecture: unfactored/factored CRBM, and style-gated unfactored/factored CRBM, we trained 21 different models on all style and speed pairs except one, which we held out for testing. Then, for each model, we attempted to predict every subsequence of length M in the test set, given the past $N = 6$ frames. We repeated the experiments for each architecture, each time reporting results averaged over the 21 models. Prediction could be performed by initializing with the previous frame and Gibbs sampling in the same way we generated, but this approach is subject to noise. We found that in all cases, integrating out the hidden units and following the gradient of the negative free energy (the log probability of an observation plus $\log Z$) with respect to $\mathbf{v}_t$ gave less prediction error. Details of how to compute the free energy by marginalizing out the binary hidden units can be found in (Freund & Haussler, 1992). The architectures were subject to different learning rates and so the number of epochs for which to train each model were determined by setting aside 10% of the training set for validation.

Fig. 6 presents the results. With almost half the number of free parameters, the 600-60 factored model performed as well as the fully parameterized CRBM. Gating with style information gives an advantage in longer-term prediction because it prevents the model from gradually changing the style. The unfactored model with style information performed slightly worse than the factored model and was extremely slow to train (it took two days to train whereas the other models were each trained in a few hours).

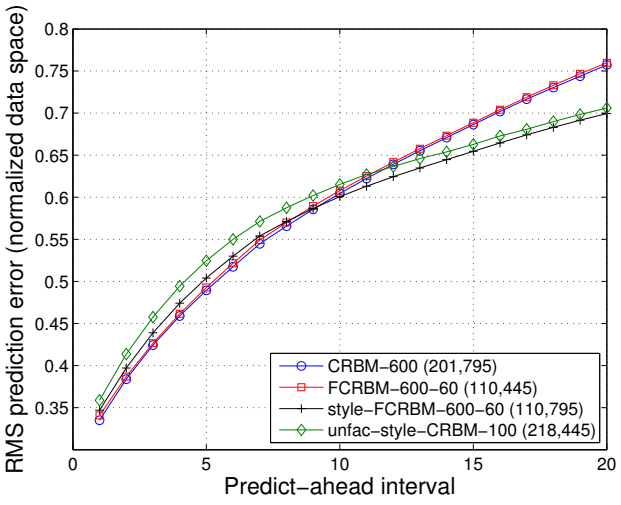


Figure 6. Prediction experiment. The number of free parameters are shown in parentheses. Error is reported in the normalized space in which the models are trained and is per-dimension, per-frame.

5. Conclusion

Restricted Boltzmann Machines have several attractive computational properties which carry through to the deeper architectures of which they form the core. From a generative model standpoint, however, these deep networks have a deficiency. Regardless of whether or not the layers below contain lateral interactions, sampling the higher layers can only determine the effective biases of the layer below. The gated CRBM is a model in which the hidden units influence the lateral interactions of the layer below, providing an exponential number of (non-independent) possible models at a cost that is cubic in the number of parameters.

If we only let contextual information (like style) determine the effective hidden biases of a CRBM, the signal is swamped by the information coming from the past. However, if we allow for a three-way, multiplicative relationship like in the gated CRBM, context becomes a natural part of the model, determining the effective weights. The potential blow-up in the number of parameters implied by such a model is solved by factoring the three-way tensors.

When modeling human motion, our approach permits style to change the effective weights of the network via discrete or real-valued representations. Changing these style-based factors during generation can induce natural-looking transitions and permit interpolation and extrapolation of styles in the training data. In our experiments we always conditioned on style, and assumed that our training data had been labeled *a priori*. This added a supervised flavour to our otherwise unsupervised models. We believe that the more interesting problem is unsupervised discovery of style.

References

Brand, M., & Hertzmann, A. (2000). Style machines. *Proc. Conf. on Comp. Graph. and Int. Techn.* (pp. 183–192).

Carreira-Perpinan, M., & Hinton, G. (2005). On contrastive divergence learning. *Proc. Int. Conf. on Artif. Intel. and Stat.* (pp. 59–66).

Freund, Y., & Haussler, D. (1992). Unsupervised learning of distributions of binary vectors using 2-layer networks. *Adv. in Neural Inf. Proc. Sys.* (pp. 912–919).

Hinton, G. (2002). Training products of experts by minimizing contrastive divergence. *Neural Comput*, *14*, 1771–1800.

Hinton, G., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Comput*, *18*, 1527–1554.

Hinton, G., & Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. *Science*, *313*, 504 – 507.

Hsu, E., Pulli, K., & Popović, J. (2005). Style translation for human motion. *Proc. Conf. on Comp. Graph. and Int. Techn.* (pp. 1082–1089).

Lawrence, N. (2004). Gaussian process latent variable models for visualisation of high dimensional data. *Adv. in Neural Inf. Proc. Sys.* (pp. 329–326).

Lee, H., Ekanadham, C., & Ng., A. (2008). Sparse deep belief net model for visual area V2. *Adv. in Neural Inf. Proc. Sys.* (pp. 873–880).

Memisevic, R. (2008). *Non-linear latent factor models for revealing structure in high-dimensional data*. Doctoral dissertation, University of Toronto.

Memisevic, R., & Hinton, G. (2007). Unsupervised learning of image transformations. *Proc. IEEE Comp. Soc. Conf. on Comp. Vis. and Pat. Rec.*.

Osindero, S., & Hinton, G. (2008). Modeling image patches with a directed hierarchy of Markov random fields. *Adv. in Neural Inf. Proc. Sys.* (pp. 1121–1128).

Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). Restricted Boltzmann machines for collaborative filtering-filtering. *Proc. Int. Conf. on Mach. Learn.* (pp. 791–798).

Salakhutdinov, R., & Murray, I. (2008). On the quantitative analysis of deep belief networks. *Proc. Int. Conf. on Mach. Learn.* (pp. 872–879).

Smolensky, P. (1986). Information processing in dynamical systems: Foundations of harmony theory. In D. E. Rumelhart, J. L. McClelland et al. (Eds.), *Parallel distributed processing: Volume 1: Foundations*, 194–281. Cambridge: MIT Press.

Taylor, G., Hinton, G., & Roweis, S. (2007). Modeling human motion using binary latent variables. *Adv. in Neural Inf. Proc. Sys.* (pp. 1345–1352).

Tieleman, T. (2008). Training restricted Boltzmann machines using approximations to the likelihood gradient. *Proc. Int. Conf. on Mach. Learn.* (pp. 1064–1071).

Torresani, L., Hackney, P., & Bregler, C. (2007). Learning motion style synthesis from perceptual observations. *Adv. in Neural Inf. Proc. Sys.* (pp. 1393–1400).

Wang, J., Fleet, D., & Hertzmann, A. (2007). Multifactor gaussian process models for style-content separation. *Proc. Int. Conf. on Mach. Learn.* (pp. 975–982).

Welling, M., Rosen-Zvi, M., & Hinton, G. (2005). Exponential family harmoniums with an application to information retrieval. *Adv. in Neural Inf. Proc. Sys.* (pp. 1481–1488).