Université de Montréal

Classification using Discriminative Restricted Boltzmann Machines

Hugo Larochelle
Yoshua Bengio

Motivation

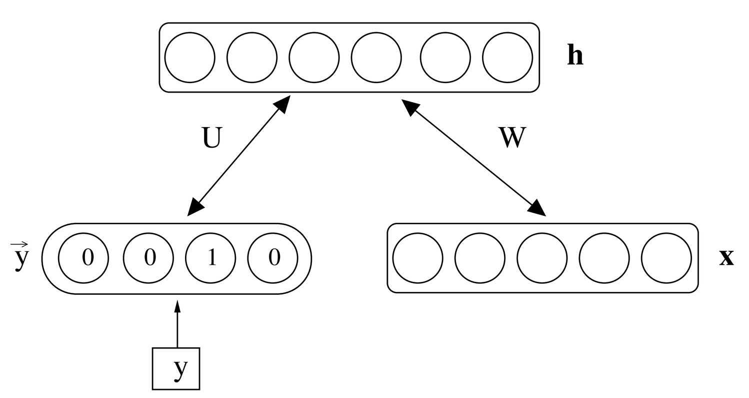
- Recently, many applications for Restricted Boltzmann Machines (RBMs) have been developed for a large variety of learning problems
- They are usually used to **extract features** or to **initialize deep neural networks**
- We argue that RBMs provide a **self-contained framework** for deriving **competitive non-linear classifiers**
- We present algorithms that **introduce a discriminative component** to RBM training
- We demonstrate how discriminative RBMs can also be successfully employed in a **semi-supervised setting**

Restricted Boltzmann Machines (RBM)

Probabilistic model over input $\mathbf{x} = (x_1, \ldots, x_d)$, target $y \in \{1, \ldots, C\}$ and binary hidden units $\mathbf{h} = (h_1, \ldots, h_n)$:

$$p(y, \mathbf{x}, \mathbf{h}) \propto \exp\left(-E(y, \mathbf{x}, \mathbf{h})\right)$$

where $E(y, \mathbf{x}, \mathbf{h}) = -\mathbf{h}^T\mathbf{W}\mathbf{x} - \mathbf{b}^T\mathbf{x} - \mathbf{c}^T\mathbf{h} - \mathbf{d}^T\vec{y} - \mathbf{h}^T\mathbf{U}\vec{y}$



- Given enough hidden units, is a **universal approximator** of distributions over a vector of binary inputs
- Conditional distributions are simple:

$$p(\mathbf{x}|\mathbf{h}) = \prod_i p(x_i|\mathbf{h}),\; p(x_i = 1|\mathbf{h}) = \mathrm{sigm}(b_i + \textstyle\sum_j W_{ji}h_j)$$

$$p(y|\mathbf{h}) = \exp\left(d_y + \textstyle\sum_j U_{jy}h_j\right) / \textstyle\sum_{y^*} \exp\left(d_{y^*} + \sum_j U_{jy^*}h_j\right)$$

$$p(\mathbf{h}|y, \mathbf{x}) = \prod_j p(h_j|y, \mathbf{x}),\; p(h_j = 1|y, \mathbf{x}) = \mathrm{sigm}(c_j + U_{jy} + \textstyle\sum_i W_{ji}x_i)$$

- Computing $p(y, \mathbf{x})$ is intractable, but it is possible to compute $p(y|\mathbf{x})$, sample from it, or choose the most probable class in $O(nd + nC)$ (Salakhutdinov, Mnih, & Hinton, 2007)

$$p(y|\mathbf{x}) = \frac{\exp(d_y) \prod_{j=1}^n \left(1 + \exp(c_j + U_{jy} + \sum_i W_{ji}x_i)\right)}{\sum_{y^*} \exp(d_{y^*}) \prod_{j=1}^n \left(1 + \exp(c_j + U_{jy^*} + \sum_i W_{ji}x_i)\right)}.$$

- Trained as generative models, i.e. maximize the **joint likelihood** of the targets and inputs, or equivalently minimize:

$$\mathcal{L}_{gen}(\mathcal{D}_{train}) = -\sum_{i=1}^{|\mathcal{D}_{train}|} \log p(y_i, \mathbf{x}_i).$$

- Training algorithm based on stochastic descent, where gradient for parameters $\theta \in \Theta$ is

$$\frac{\partial \log p(y_i, \mathbf{x}_i)}{\partial \theta} = -\mathbf{E}_{\mathbf{h}|y_i, \mathbf{x}_i}\left[\frac{\partial}{\partial \theta} E(y_i, \mathbf{x}_i, \mathbf{h})\right] + \mathbf{E}_{y, \mathbf{x}, \mathbf{h}}\left[\frac{\partial}{\partial \theta} E(y, \mathbf{x}, \mathbf{h})\right]$$

and is estimated using **Contrastive Divergence**

Discriminative Restricted Boltzmann Machines (DRBM)

- In a classification setting, we are not interested in obtaining a good model of the input distribution $p(\mathbf{x})$. It can be advantageous to maximize the **conditional likelihood** of the targets, or equivalently minimize:

$$\mathcal{L}_{disc}(\mathcal{D}_{train}) = -\sum_{i=1}^{|\mathcal{D}_{train}|} \log p(y_i|\mathbf{x}_i)$$

- DRBM could be trained by Contrastive Divergence too, however exact gradient can be computed:

$$\frac{\partial \log p(y_i|\mathbf{x}_i)}{\partial \theta} = \sum_j \mathrm{sigm}(o_{yj}(\mathbf{x}_i)) \frac{\partial o_{yj}(\mathbf{x}_i)}{\partial \theta} - \sum_{j, y^*} \mathrm{sigm}(o_{y^*j}(\mathbf{x}_i)) p(y^*|\mathbf{x}_i) \frac{\partial o_{y^*j}(\mathbf{x}_i)}{\partial \theta}$$

where $o_{yj}(\mathbf{x}) = c_j + \sum_k W_{jk}x_k + U_{jy}$

- Using this gradient, we can perform stochastic gradient descent

Contrastive Divergence

Algorithm for Contrastive Divergence parameter update

Input: training pair $(y_i, \mathbf{x}_i)$ and learning rate λ
% Notation: $a \leftarrow b$ means a is set to value b
% $a \sim p$ means a is sampled from p

% Positive phase
$y^0 \leftarrow y_i$, $\mathbf{x}^0 \leftarrow \mathbf{x}_i$, $\widehat{\mathbf{h}}^0 \leftarrow \mathrm{sigm}(\mathbf{c} + W\mathbf{x}^0 + \mathbf{U}\vec{y}^0)$

% Negative phase
$\mathbf{h}^0 \sim p(\mathbf{h}|y^0, \mathbf{x}^0)$, $y^1 \sim p(y|\mathbf{h}^0)$, $\mathbf{x}^1 \sim p(\mathbf{x}|\mathbf{h}^0)$
$\widehat{\mathbf{h}}^1 \leftarrow \mathrm{sigm}(\mathbf{c} + W\mathbf{x}^1 + \mathbf{U}\vec{y}^1)$

% Update
for $\theta \in \Theta$ **do**
$\theta \leftarrow \theta - \lambda\left(\frac{\partial}{\partial\theta}E(y^0, \mathbf{x}^0, \widehat{\mathbf{h}}^0) - \frac{\partial}{\partial\theta}E(y^1, \mathbf{x}^1, \widehat{\mathbf{h}}^1)\right)$
end for

Hybrid Discriminative Restricted Boltzmann Machines (HDRBM)

- The advantage brought by discriminative training depends on the amount of available training data. Smaller training sets favor generative learning, bigger ones favor discriminative learning
- Instead of solely relying on just one perspective, we can adopt a hybrid discriminative/generative approach simply by **combining the respective training criteria**

$$\mathcal{L}_{hybrid}(\mathcal{D}_{train}) = \mathcal{L}_{disc}(\mathcal{D}_{train}) + \alpha\mathcal{L}_{gen}(\mathcal{D}_{train})$$

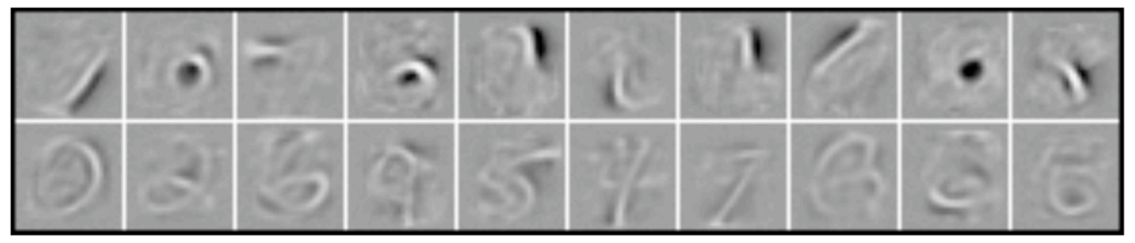
- To train HDRBM, we use stochastic gradient descent and add gradient contribution due to $\mathcal{L}_{disc}$ with α times the gradient estimator for $\mathcal{L}_{gen}$

Character Recognition

- Experiment on the MNIST dataset (50000, 10000 and 10000 example in training, validation and test sets)
- Sparse version of HDRBM: push biases of hidden units down by subtracting δ after every parameter update

Model	Error
RBM ($\lambda = 0.005$, $n = 6000$)	3.39%
DRBM ($\lambda = 0.05$, $n = 500$)	1.81%
RBM+NNet	1.41%
HDRBM ($\alpha = 0.01$, $\lambda = 0.05$, $n = 1500$)	1.28%
Sparse HDRBM (idem + $n = 3000$, $\delta = 10^{-4}$)	**1.16%**
SVM	1.40%
NNet	1.93%

Subset of filters learned by the HDRBM on the MNIST dataset



Some filters act as edge detectors (first row), some other filters are more specific to a particular digit shape (second row)

Semi-supervised Learning in RBMs

- What about a classification setting where there are few labeled training data but many unlabeled examples of inputs?
- Semi-supervised learning algorithms address this situation by using the unlabeled data to introduce constraints on the trained model
- In the RBM framework, a natural constraint is to **ask model to be a good generative model of unlabeled data**:

$$\mathcal{L}_{unsup}(\mathcal{D}_{unlab}) = -\sum_{i=1}^{|\mathcal{D}_{unlab}|} \log p(\mathbf{x}_i), \text{ where } \mathcal{D}_{unlab} = \{(\mathbf{x}_i)\}_{i=1}^{|\mathcal{D}_{unlab}|}$$

- Contrastive Divergence can also be used to estimate the likelihood gradient:

$$\frac{\partial \log p(\mathbf{x}_i)}{\partial \theta} = -\mathbf{E}_{y, \mathbf{h}|\mathbf{x}_i}\left[\frac{\partial}{\partial \theta} E(y_i, \mathbf{x}_i, \mathbf{h})\right] + \mathbf{E}_{y, \mathbf{x}, \mathbf{h}}\left[\frac{\partial}{\partial \theta} E(y, \mathbf{x}, \mathbf{h})\right]$$

- Need only to change statement $y^0 \leftarrow y_i$ with $y^0 \sim p(y|\mathbf{x}_i)$ in previous algorithm
- To perform semi-supervised learning, weight and combine the $\mathcal{L}_{unlab}$ with $\mathcal{L}_{gen}$, $\mathcal{L}_{disc}$ or $\mathcal{L}_{hybrid}$
- Classification performance comparison with standard non-parametric semi-supervised algorithm based on function induction (Bengio, Delalleau, & Le Roux, 2006):

Model	MNIST	MNIST-BI	20-news
HDRBM	9.73%	42.4%	40.5%
Semi-sup HDRBM	8.04%	**37.5%**	**31.8%**
NP-Gauss	10.60%	66.5%	85.0%
NP-Trunc-Gauss	**7.49%**	61.3%	82.6%

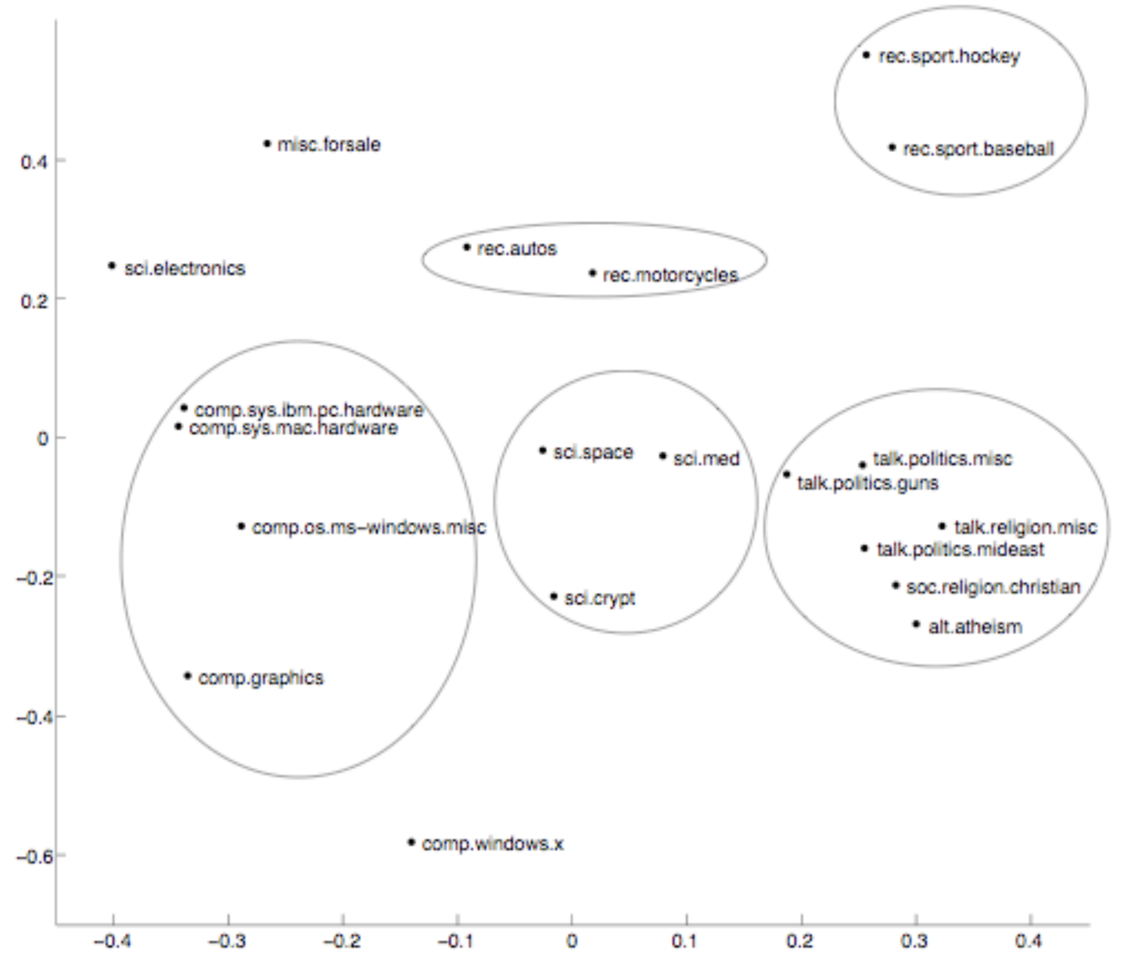
- Outperforms SVM **trained on full 20newsgroup training set**!

Document Classification

- Experiment on the 20newsgroup dataset (5000 most frequent words, with 9578, 1691 and 7505 examples in training, validation and test sets)

Model	Error
RBM ($\lambda = 0.0005$, $n = 1000$)	24.9%
DRBM ($\lambda = 0.0005$, $n = 50$)	27.6%
RBM+NNet	26.8%
HDRBM ($\alpha = 0.005$, $\lambda = 0.1$, $n = 1000$)	**23.8%**
SVM	32.8%
NNet	28.2%

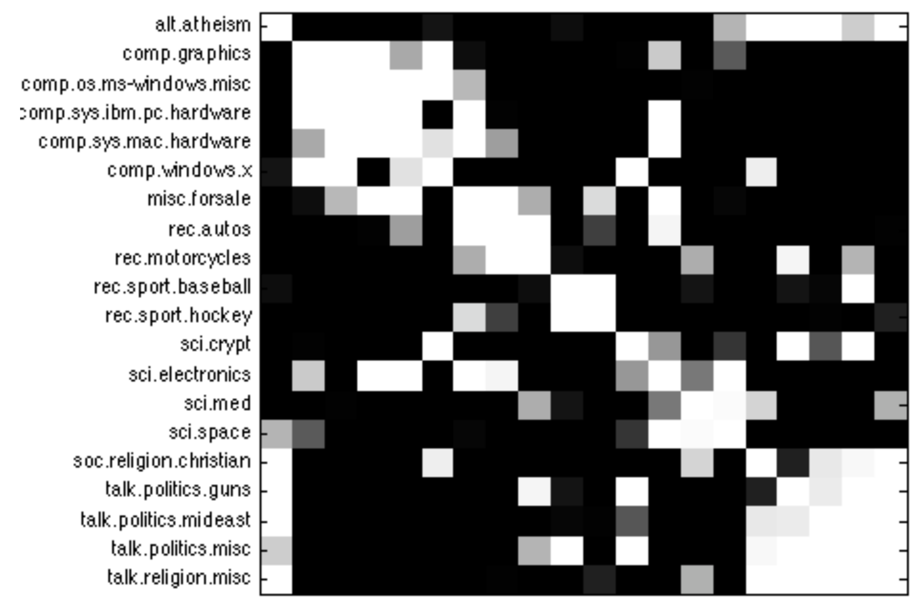
Two dimensional PCA embedding of the newsgroup weights



Most influential words per document newsgroup

Class	Words
alt.atheism	bible, atheists, benedikt, atheism, religion, scholars
comp.graphics	tiff, ftp, window, gif, images, pixel, rgb, viewer, image
comp.os.ms-windows.misc	windows, cica, bmp, window, win, installed, toronto, dos
comp.sys.ibm.pc.hardware	dos, ide, adaptec, pc, config, irq, vlb, bios, scsi, esdi, dma
comp.sys.mac.hardware	apple, mac, quadra, powerbook, lc, pds, centris, fpu
comp.windows.x	xlib, man, motif, widget, openwindows, xterm, colormap
misc.forsale	sell, condition, floppy, week, am, obo, shipping, company
rec.autos	cars, ford, autos, sho, toyota, roads, vw, callison, sc, drive
rec.motorcycles	bikes, motorcycle, ride, bike, dod, rider, bmw, honda
rec.sport.baseball	pitching, braves, hitter, ryan, pitchers, so, rbi, yankees
rec.sport.hockey	playoffs, penguins, didn, playoff, game, out, play, cup
sci.crypt	sternlight, bontchev, nsa, escrow, hamburg, encryption
sci.electronics	amp, cco, together, voltage, circuits, detector, connectors
sci.med	drug, syndrome, dyer, diet, foods, physician, medicine
sci.space	orbit, spacecraft, speed, safety, known, lunar, then, rockets
soc.religion.christian	rutgers, athos, jesus, christ, geneva, clh, christians, sin
talk.politics.guns	firearms, handgun, firearm, gun, rkba, concealed, second
talk.politics.mideast	armenia, serdar, turkish, turks, cs, argic, stated, armenians
talk.politics.misc	having, laws, clinton, time, koresh, president, federal
talk.religion.misc	christians, christian, bible, weiss, religion, she, latter

Similarity matrix of the newsgroup weights vectors $U_{\cdot y}$



Future Work:

- Investigate the use of discriminative versions of RBMs in more challenging settings such as in multi-task or structured output problems
- Explore ways to introduce generative learning in RBMs and HDRBMs which would be less computationally expensive when the input vectors are large but sparse

References

Bengio, Y., Delalleau, O., & Le Roux, N. (2006). Label propagation and quadratic criterion. In Chapelle, O., Schölkopf, B., & Zien, A. (Eds.), *Semi-Supervised Learning*, pp. 193–216. MIT Press.

Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). Restricted boltzmann machines for collaborative filtering. In *ICML '07: Proceedings of the 24th international conference on Machine learning*, pp. 791–798 New York, NY, USA. ACM.