Sparse Deep Belief Net for Handwritten Digits Classification

Jiongyun Xie[1], Hongtao Lu[2], Deng Nan[3], and Cai Nengbin[4]

[1,2] MOE-MS Key Laboratory for Intelligent Computing and Intelligent Systems
Dept. of Computer Science, Shanghai Jiaotong University, Shanghai, China
andingxie@gmail.com, lu-ht@cs.sjtu.edu.cn
[3,4] Shanghai Forensic Center
{dengnan1985,cainengbin}@sina.cn

Abstract. It has been shown that the Deep Belief Network is good at modeling input distribution, and can be trained efficiently by the greedy layer-wise unsupervised learning. Hoglak Lee et al. (2008) introduced a sparse variant of the Deep Belief Network, which applied the Gaussian linear units to model the input data with a sparsity constraint. However, it takes much more weight updates to train the RBM (Restricted Boltzmann Machine) with Gaussian visible units, and the reconstruction error is much larger than training an RBM with binary visible units. Here, we propose another version of Sparse Deep Belief Net which applies the differentiable sparse coding method to train the first level of the deep network, and then train the higher layers with RBM .This hybrid model, combining the advantage of the Deep architecture and the sparse coding model, leads to state-of-the-art performance on the classification of handwritten digits.

Keywords: Deep Belief Network, Restricted Boltzmann Machine, Sparse Coding.

1 Introduction

Restricted Boltzmann Machine[5] is a powerful generative model to model the data distribution.In earlier training method,prolong Gibbs sampling method was proposed to obtain the equilibrium distribution of the network.However, this was very time-consuming and samples from the equilibrium distribution generally have high variance since they come from all over the models distribution.Hinton et al.[5] propose a contrastive divergence method to train the RBM.Instead of running the Gibbs sampler to reach its equilibrium, their method needs only one Gibbs sampling iteration to approximate the data reconstruction.Although it is not exactly following the gradient of the log probability of the training data,contrastive-divergence has been shown to produce only a small bias for a large speed-up in training time,and the variance coming from the reconstruction is very small because the one-step reconstruction is very similar to the data.

The contrastive-divergence learning algorithm makes the RBM available to practical applications. Hinton et al.[4] proposed a greedy layer-wise training

F.L. Wang et al. (Eds.): AICI 2010, Part I, LNAI 6319, pp. 71–78, 2010.
© Springer-Verlag Berlin Heidelberg 2010

method to train the deep belief net.RBM was used here to train every layer of the deep architecture to provide the initial value for the whole network,then the deep network can be fine-tuned with supervised back-propagation.It performs excellent on the handwritten digits classification.

Hoglak Lee et al.[6] proposed a sparse RBM which applies the Gaussian(linear) visible units to model the input data and adds a regularization term that penalizes a derivation of the expected activation of the hidden units from a (low) fixed level p to produce sparse codes.As long as the p is sufficient small,the sparse RBM will learn the sparse (stroke like) features from modeling the input data as other sparse coding model. To train the sparse RBM,one needs only to apply the contrastive divergence update rule,followed by one step of gradient descent using the gradient of the regularization term.

Nonetheless,learning an RBM with Gaussian visible units can be slow, as it may require a much grater number of weight updates than an equivalent RBM with binary visible units. Moreover,the reconstruction error in training the RBM with Gaussian visible units is much large compared with the equivalent RBM with binary visible units.So we conjecture that contrastive-divergence may not be accurate enough to train the RBM with Gaussian units.

In this paper, we develop a different version of the sparse deep belief net.We train the first layer of deep belief net with the differentiable sparse coding scheme[13] instead of RBM.Then we take the sparse codes as the input to the higher layer of the deep belief network.Because the sparse codes produced by the sparse coding scheme are quasi binary,we can train the higher layer of the deep belief net with the standard binary RBM perfectly.

After the pre-training strategy,we obtained a sparse Deep Belief Net which keeps the powerful information of the sparse features from modeling the input data.We will show that this sparse Deep Belief Net perform very well on the discriminative task.

2 Restricted Boltzman Machine

Restricted Boltzmann Machine (RBM) is an undirected graphical model (Markov random field).It is a powerful generative model with one visible layer, one hidden layer and no intra-layer connections.

The energy function of RBM is

$$E(v,h) = -\sum_i v_i\theta_i - \sum_j h_j\lambda_j - \sum_{ij} W_{ij}v_ih_j \tag{1}$$

where the v and h represent the corresponding visible units and the hidden units.W is the matrix of the pair-wise weights.The corresponding joint distribution for the RBM is

$$p(v,h) = \frac{exp(-E(v,h))}{E_I} \tag{2}$$

where

$$E_I = \sum_{u,g} exp(-E(u,g)) \tag{3}$$

is the partition function of the model. From (2) we can compute

$$p(v) = \frac{1}{E_I} \cdot \sum_h (exp(-E(v,h))) \tag{4}$$

Given a set of N training cases $v_1, v_2, v_3 ... v_n$, we then train the parameter by maximizing

$$L = \sum_{i=1}^{n} \log P(v_i) \tag{5}$$

so as to minimize the energy of states drawn from the data distribution and raise the energy of states that are improbable given the data. To calculate the gradient of L with respect to the parameter θ, we obtain

$$\frac{\partial L}{\partial \theta} = -\langle \frac{\partial E}{\partial \theta} \rangle_{p_0(h,v)} + \langle \frac{\partial E}{\partial \theta} \rangle_{p_\infty(h,v)} \tag{6}$$

where p_∞ represents the equilibrium distribution of the model,which can be obtained by running prolong Gibbs sampling, p_0 represents the distribution of the model when the visible units are clamped on the training data.However, to get the equilibrium distribution is very time-consuming and would produce samples with high variance.Hinton et.al[5] introduced the contrastive divergence which has been found to be efficient in training the energy-based model.The idea is that instead of running the Gibbs sampler to its equilibrium distribution, they get the samples by running only one (or a few) Gibbs sampling.

2.1 Sparse RBM with Gaussian Visible Units

Welling et.al[9] introduced Gaussian hidden units which were used in an information retrieval task. Here we are interested in applying the linear units to handle with the continuous input. The energy function for an RBM with Gaussian visible unit is

$$E(v,h) = \sum_{i \in pixels} \frac{(v_i - b_i)^2}{2{\sigma_i}^2} - \sum_{j \in features} b_j h_j - \sum_{i,j} \frac{v_i}{\sigma_i} h_j u_{ij} \tag{7}$$

where σ_i is the standard deviation of the Gaussian noise for unit i. The update rule is similar to the binary case.When the linear units have Gaussian noise with unit variance,the stochastic undate rule for the hidden unit remain the same,while the visible units i is to sample from a Gaussian with mean $b_i + \sum_j h_j W_{ij}$ and unit variance.When the variance is not 1,it need only minor adjustment which can be found in [4].

Hoglak Lee et.al[6] proposed a sparse version of the RBM which apply the Gaussian(linear) units to model the input data and add a regularization term to penalize the derivation of the expected activation of the hidden units from a(low) fixed level p, as follows:

$$\max\{\log P(v) - \gamma(p - E(h|v))\} \tag{8}$$

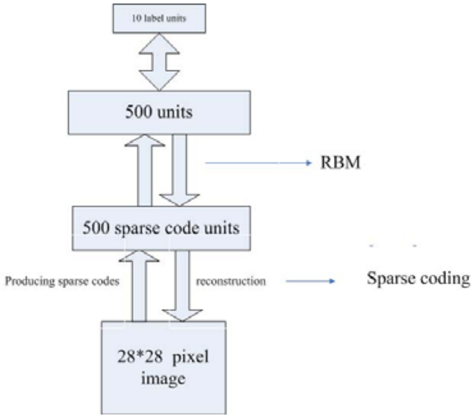


Fig. 1. The framework of our sparse DBN model. The first layer of the net is initialized with the sparse coding algorithm,and the higher layer is trained with RBM.

where v represents the training data,$E(h|v)$ is the expect activation of the hidden units.The training algorithm needs only small modification which can be found in [6].

However, the contrastive divergence learning may not be powerful enough to train RBM with Gaussian linear visible units.Compared with the RBM with binary units,the reconstruction error is much larger. We analyze this as follows, when trained with the binary units,the RBM only needs to model the input distribution that scaled in (0,1). However, when trained with the continuous valued inputs,the RBM has to model the unconstrained valued data distribution,and the energy surface is much more sophisticate,so it is much more difficult to model the manifold of and around the training samples.So we conjecture that the RBM with Gaussian linear visible units trained with contrastive divergence may not be sufficient accurate to be applied in a discriminative task.In the following section we propose an sparse coding strategy to train the first linear layer of the deep nets.

3 Sparse Deep Belief Net

Motivated by the sparse RBM,we propose an differentiable sparse-coding scheme that was used in [13] to train the first layer of the deep net.Then we take the sparse codes as the input of the higher layer,and train the weights of the higher layer with the standard RBM(see Figure 1).

3.1 Differentiable Sparse Coding

In sparse methods,the codes is forced to have only a few non-zero units while most units are zeros or close to zero most of the time.It has been shown that sparse over-complete representation have many theoretical and practical advantages,as is shown in [2][7].In particular,sparse codes have good robustness to noise and perform well in the classification tasks.Sparse representation is motivated by the organization of the cortex.The receptive fields in visual area V1 have been reasonably well described physiologically and can be characterized as being localized,oriented and bandpass which is similar to the filters learned from sparse coding.

The differentiable sparse-coding scheme[13] used here is to minimize the the following equation,with respect to the sparse codes W,

$$Loss = \sum_i \{(x_i - BW_i)^2 + \gamma D_p(W_i||p)\} \tag{9}$$

where x represents the input data and B is the basis of the model.$D_p(w||p)$ is the regularization function which measure the distance between the sparse code w and a parameter vector p,here we use the unnormalized KL-divergence[13]:

$$D_p(w||p) = \sum_j (w_j \log \frac{w_j}{p_j} - w_j + p_j) \tag{10}$$

When the constant vector p is sufficient close to zero,KL-divergence prior can approximate the L_1 prior[13]. L_1 prior has been shown to be good at producing sparse codes.However,L_1 prior does not produce differentiable MAP(Maximum-A-Posteriori) estimates.KL-divergence prior,which preserves the sparsity benefits of the L_1-regularization,is smooth that can produce much stable latent codes which lead to better classification performance. Additionally,because of the smoothness of the KL-divergence prior,the basis B,can be optimized discriminatively through gradient descent back-propagation[13].

3.2 Combining Deep Belief Net and Sparse Coding

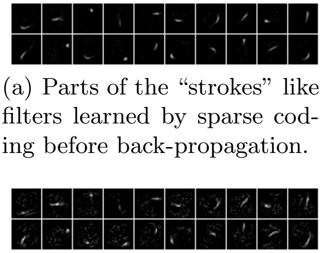
After training the first layer of the deep belief net using differentiable sparse coding,it can be easily integrated in the Deep Belief Net.We take the sparse codes as the input of the higher layer,then train the weights of the higher layer with the standard RBM.We can model the codes with the standard RBM perfectly because the codes we obtain are sparse and quasi binary.

To apply the Deep Belief Net in a classification task,unsupervised training is not enough.After training the hybrid model described above to initialize the weights of the deep belief net,we fine-tune the whole network with back-propagation. Because of the smooth KL-divergence prior,the gradient of the sparse codes w with respect to the basis B can be computed with the implicit differentiation,which can be found in[13].

4 Experiments

4.1 Learning the Sparse Feature from Handwritten Digits

We applied the differentiable sparse coding scheme to the MNIST handwritten digit data set.Here we separated the original training set into training and



(a) Parts of the “strokes” like filters learned by sparse coding before back-propagation.

(b) The corresponding filters of the sparse-coding layer of the DBN after back-propagation.

Fig. 2. From the above two graph,we can see that even after the backpropagate,the first layer of the Deep Belief Network keeps most of the information learned from the sparse coding algorithm.

Table 1. Comparison of the classification performances on the MNIST database.Our sparse version of deep belief net(sparse coding+RBM) achieves the best result.SVM and multilayer neural network results are taken from http://yann.lecun.com/exdb/mnist/.On this data set,differences of 0.2% in classification error is statistically significant.

Model	Error
RBM+RBM	1.35%
Sparse RBM+RBM	1.42%
Sparse Coding+RBM	1.33%
Multilayer neural network	1.51%
SVM	1.40%

validation sets of 50000 and 10000 examples and used the standard test set of 10000 examples.We learned a sparse layer with 784 visible units and 500 hidden units.The learned bases are shown in Figure 2(a).In Figure 2(a) we can see that the basis found by the model roughly represent different strokes of which handwritten digits are comprised.This is consistent with results obtained by applying different sparse coding algorithms to learn sparse representations of this handwritten data set.

4.2 Learning the Sparse Deep Belief Net

By applying the 500 sparse codes obtained from the above algorithm as the input to the higher layer,we used the RBM model to train the higher layer of the net with 500 hidden units.The layer-by-layer pre-training provided us with the initial value of the whole network.At the fine-tuning stage,we added an extra layer with 10 units for the digit classification and proposed the conjugate gradient back-propagation to minimize the cross-entropy error function.Then we obtained a 784-500-500-10 multilayer network.The experiment results are given in table 1.The RBM+RBM uses the standard RBM to initialize the whole Deep Belief network.Sparse RBM+RBM uses the sparse RBM to train the first layer of the Deep Belief Network,and sparse coding+RBM is the model we propose in this paper. We give as a comparison the results of a Gaussian kernel SVM and a regular neural network without pre-training.

We observe that the Deep Belief Networks with pretraining performs better than the regular neural network(random initialization),and our Sparse Deep Belief Network outperforms other DBN models.

Pre-training helps generalization because it ensures that most of the information in the initial weights come from modeling the handwritten digits.In our experiment,the filters learned in the sparse layer before and after fine-tuning are shown in Figure 2,from which one can see that back-propagation hasnt destroyed the sparse(stroke like) features. So even after the back-propagation,the network still keeps most of the information from modeling the input images.This explains why the Deep Belief Net model with pre-training perform better than the neural network without pre-training.

Our sparse version of Deep Belief Net achieves the best result on the MNIST handwritten data set,which implies the advantage of the sparse features on classification.It has been shown in many papers that sparse coding perform excellent on both the generative and discriminative models[12][13].The sparse codes are much more robust than the original input. The sparse features represent the deep structure of the input data.

5 Conclusion

This paper is motivated by the need to develop good training algorithms for deep architectures.It has been show that layer-by-layer training would improve the generative power of the deep architectures as long as the number of the hidden units are sufficient large[1].Deep Belief Networks and the sparse coding models have many characters in common.They both exploit the deep structure of the input data.In this paper,we combine the advantage of Deep Belief Networks and Sparse coding together and construct a discriminative model with quiet good result on the classification of the handwritten digit data set.

For future work,we would like to investigate the model in various task,including the facial feature extraction and classification.We are also interested in applying the hybrid model in the generative task[13][14].Deep Belief Networks have been shown to have powerful generative ability to model the distribution of the input data.If we combine it with the superior sparse features,maybe we will make fundamental improvement.

Acknowledgments. We thank Heng Luo for helpful discussions on Deep Belief Networks.Thanks to Huiyan Tu for discussions on the Sparse Coding. This work is supported by Shanghai Committee of Science and Technology (No.08JG05002) and 863 (No. 2008AA02Z310).

References

1. Hinton, G.E., Osindero, S., Yee-Whye, T.: A Fast Learning Algorithm for Deep Belief Nets. Neural Computation 18, 1527–1544 (2006)
2. Olshausen, B.A.: Emergence of Simple-cell Receptive Field Properties by Learning a Sparse Code for Natural images. Nature 381, 607–609 (1996)
3. Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H.: Greedy Layer-Wise Training of Deep Networks. In: NIPS (2007)
4. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science 313(5786), 504–507 (2006)
5. Hinton, G.E.: Training Products of Experts by Minimizing Contrastive Divergence. Neural Computation 14, 1771–1800 (2002)
6. Lee, H., Ekanadham, C., Ng, A.Y.: Sparse Deep Belief Net Model for Visual Area V2. In: NIPS (2008)
7. Ranzato, M., Poultney, C., Chopra, S., LeCun, Y.: Efficient Learning of Sparse Representations with an Energy-based Model. In: NIPS (2006)
8. Ranzato, M., Boureau, Y.-L., LeCun, Y.: Sparse Feature Learning for Deep Belief Networks. In: NIPS (2007)
9. Welling, M., Rosen Zvi, M., Hinton, G.E.: Exponential Family Harmoniums with an Application to Information Retrieval. In: NIPS (2005)
10. Teh, Y.W., Welling, M., Osindero, S., Hinton, G.E.: Energy-based Models for Sparse Overcomplete Representations. Journal of Machine Learning Research 4, 1235–1260 (2003)
11. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient Sparse Coding Algorithm. In: NIPS (2006)
12. Raina, R., Battle, A., Lee, H., Packer, B., Ng, A.Y.: Self-taught Learning: Transfer Learning from Unlabeled Data. In: ICML (2007)
13. Bradley, D.M., Bagnell, D.: Differentiable Sparse Coding. In: NIPS (2008)
14. Larochelle, H., Bengio, Y.: Classification using Discriminative Restricted Boltzmann Machine. In: ICML (2008)
15. Nair, V., Hinton, G.: Implicit Mixtures of Restricted Boltzmann Machine. In: NIPS (2008)
16. The MNIST database of handwritten digits, `http://yann.lecun.com/exdb/mnist/`