Method to Improve the Performance of Restricted Boltzmann Machines

Jing Yin[1,2], Qingyu Mao[3], Dayiheng Liu[1], Yong Xu[1], and Jiancheng Lv[1(✉)]

[1] Machine Intelligence Laboratory, College of Computer Science, Sichuan University, Chengdu 610065, People's Republic of China
lvjiancheng@scu.edu.cn
[2] College of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, People's Republic of China
[3] Archives, Sichuan University, Chengdu 610065, People's Republic of China

Abstract. Restricted Boltzmann machines (RBMs) are widely applied to solve many machine learning problems. Usually, the cost function of RBM is log-likelihood function of marginal distribution of input data, and the training method involves maximizing the cost function. Distribution of the trained RBM is identical to that of input data. But the reconstruction error always exists even the distributions are almost identical. In this paper, a method to train RBM by adding reconstruction error to the cost function is put forward. Two categories of trials are performed to validate the proposed method: feature extraction and classification. The experimental results show that the proposed method can be effective.

Keywords: Restricted Boltzmann machine · Feature learning
Reconstruction error · Classification

1 Introduction

Restricted Boltzmann machines (RBMs) [1] have been successfully used to many tasks of machine learning, including collaborative filtering [2], feature extraction [3], dimensionality reduction [4], object recognition [5], classification [6], and many others. RBMs usually extract features by unsupervised learning. The RBMs could be initializers of other neural networks [7], solve classification problems with other classifiers [7,8], or form deep belief nets (DBNs) [9] and deep Boltzmann machines (DBMs) [10].

RBM is an undirected graph model based on energy function, which consists of two layers. The training objective of RBM is maximizing the log-likelihood function of marginal distribution of input data. When the distribution learned by RBM is identical to the distribution of input data, the training is complete. However, reconstruction error always exists even if the distributions are almost identical. So, a method which adds reconstruction error to the cost function of RBM is presented to improve the performance of RBM. In fact, there are

T. Huang et al. (Eds.): ISNN 2018, LNCS 10878, pp. 115–123, 2018.
https://doi.org/10.1007/978-3-319-92537-0_14

some literatures that use reconstruction error to improve the performance of RBM [11–15]. [11] uses reconstruction error as the criterion for cutting down the learning rate. [12] proposes an approach for RBM training. The approach used a normalized reconstruction error to determine increment necessity and compute the number of additional features for the increment. [13] proposes a new training technique for deep belief neural network, which based on minimizing the reconstruction error. [14] trains a new model by selecting a subset of the training set through reconstruction errors. [15] trains a new model by using reconstruction errors themselves. However, we use the reconstruction error as the part of the cost function of RBM. In the case of ensuring that the distribution learned by the model is identical to the distribution of input data, the reconstruction error is as small as possible to achieve better performance. We make experiments on several public databases to verify the effectiveness of the proposed method. Compared with RBM, the proposed method could be better on feature extraction and classification.

In the rest of this paper, we give an outline of the RBM in Sect. 2, introduce the proposed method in Sect. 3, implement several experiments and analyze the experimental results in Sect. 4, and provide the conclusion in final section.

2 Restricted Boltzmann Machines

RBM is a random neural network model, which consists of two layers: visible layer and hidden layer shown in Fig. 1. Visible layer with $|\mathbf{v}|$ neurons represents input data, and hidden layer with $|\mathbf{h}|$ neurons is representation of the input data. $\mathbf{W}$ is the connections weight between the visible layer and the hidden layer.

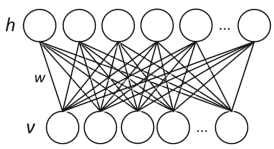

Fig. 1. Restricted Boltzmann machine.

Energy function of RBM takes following form:

$$E(\mathbf{v}, \mathbf{h}|\theta) = -\sum_{m=1}^{|\mathbf{v}|} a_m v_m - \sum_{n=1}^{|\mathbf{h}|} c_n h_n - \sum_{m=1}^{|\mathbf{v}|} \sum_{n=1}^{|\mathbf{h}|} W_{mn} h_n v_m, \tag{1}$$

where θ denotes the real-valued parameters a_m, c_n and W_{mn}, and $v_m \in \{0, 1\}$, $h_n \in \{0, 1\}$. According to the energy function, the joint distribution of the RBM is defined by

$$p(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v}, \mathbf{h})}, \tag{2}$$

where $Z = \sum_{\mathbf{v},\mathbf{h}} e^{-E(\mathbf{v},\mathbf{h})}$ is a normalization constant. Conditioned on $\mathbf{v}$, the probability of hidden neuron n with the value of 1 has the form

$$p(h_n = 1|\mathbf{v}) = \sigma(c_n + \sum_{m=1}^{|\mathbf{v}|} W_{mn} v_m), \tag{3}$$

where $\sigma(y) = 1/(1+e^{-y})$ is the logistic sigmoid function. Conditioned on $\mathbf{h}$, the probability of visible neuron m with the value of 1 has the form

$$p(v_m = 1|\mathbf{h}) = \sigma(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn} h_n), \tag{4}$$

Given the marginal probability $p(\mathbf{v})$, the cost function of the RBM is given by $L(\theta) = \frac{1}{|T|} \sum_{t=1}^{|T|} logp(\mathbf{v}^{(t)}; \theta)$, and θ could be optimized by gradient ascent on the log-likelihood. $|T|$ is the quantity of training data. In this formula, calculating partial derivative of $l(\theta) = logp(\mathbf{v}^{(t)}; \theta)$ is the key. For any input data $(\mathbf{v}^{(t)})$, the gradient of θ has the form:

$$\frac{\partial l(\theta)}{\partial \theta} = -\sum_{\mathbf{h}} p(\mathbf{h}|\mathbf{v}^{(t)}) \frac{\partial E(\mathbf{v}^{(t)}, \mathbf{h}|\theta)}{\partial \theta} + \sum_{\mathbf{v},\mathbf{h}} p(\mathbf{v},\mathbf{h}) \frac{\partial E(\mathbf{v},\mathbf{h}|\theta)}{\partial \theta}. \tag{5}$$

From Eq. 5, partial derivative of the parameter W_{mn} can be obtained by

$$\frac{\partial l(\theta)}{\partial W_{mn}} = p(h_n = 1|\mathbf{v}^{(t)}) v_m^{(t)} - \sum_{\mathbf{v}} p(\mathbf{v}) p(h_n = 1|\mathbf{v}) v_m. \tag{6}$$

Because of the existence of normalization constant $Z(\theta)$, the computational complexity of the gradient is very high. In order to reduce the computational complexity, approximate calculations are usually used, such as the contrastive divergence (CD) [16] algorithm. The connection weight $\mathbf{W}$ learns the features of the input data, and its gradients are relevant to the probability of $\mathbf{h}$. Given $\mathbf{h}$, we can compute the active probability of $\mathbf{v}$, then get reconstructions by sampling.

3 Improved Training Method

The objective of RBM is updating model parameters to make model distribution and input data distribution as identical as possible. In fact, the difference between the input data and the reconstructions always exists, even if the distributions are almost identical. The difference is called reconstruction error, we define the reconstruction error as $\varepsilon = \|\mathbf{v}' - \mathbf{v}\|^2$, where $\mathbf{v}$ is any one of the input data, $\mathbf{v}'$ is a reconstruction of $\mathbf{v}$. Here we propose a method to make the distributions as identical as possible, while the reconstruction error as small as possible. The basic idea of the improved training method is to add the reconstruction

error into the cost function of RBM and generate a new cost function. The new cost function of RBM could be defined as

$$L(\theta) = \frac{1}{|T|}\sum_{t=1}^{|T|}\left(logp(\mathbf{v}^{(t)};\theta) - \frac{1}{2}\|\mathbf{v}' - \mathbf{v}^{(t)}\|^2\right), \tag{7}$$

where $\mathbf{v}'$ can be computed by $v'_m = \sigma(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n)$. In order to distinguish, RBM with the new cost function is called reRBM. To train the reRBM, we should maximize $L(\theta)$. The same as RBM, we define $l(\theta) = logp(\mathbf{v}^{(t)};\theta) - \frac{1}{2}\|\mathbf{v}' - \mathbf{v}^{(t)}\|^2$, the updating formula of θ is:

$$\frac{\partial l(\theta)}{\partial\theta} = -\sum_{\mathbf{h}} p(\mathbf{h}|\mathbf{v}^{(t)})\frac{\partial E(\mathbf{v}^{(t)},\mathbf{h}|\theta)}{\partial\theta} + \sum_{\mathbf{v},\mathbf{h}} p(\mathbf{v},\mathbf{h})\frac{\partial E(\mathbf{v},\mathbf{h}|\theta)}{\partial\theta} - (\mathbf{v}' - \mathbf{v}^{(t)})\frac{\partial(\mathbf{v}' - \mathbf{v}^{(t)})}{\partial\theta}. \tag{8}$$

The reRBM uses CD-1 algorithm similar to RBM does. Specifically, the updating formulas of the parameters W_{mn}, c_n and a_m are:

$$W_{mn} = W_{mn} + \epsilon(p(h_n = 1|\mathbf{v}^{(t)})v_m^{(t)} - p(h'_n = 1|\mathbf{v}')v'_m - (v'_m - v_m^{(t)})\dot{\sigma}(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n)h_n), \tag{9}$$

$$a_m = a_m + \epsilon((v_m^{(t)} - v'_m) - (v'_m - v_m^{(t)})\dot{\sigma}(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n)), \tag{10}$$

$$c_n = c_n + \epsilon(p(h_n = 1|\mathbf{v}^{(t)}) - p(h'_n = 1|\mathbf{v}')). \tag{11}$$

where $\dot{\sigma}(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n) = \sigma(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n)[1 - \sigma(a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn}h_n)]$, ϵ is the learning rate of the reRBM. The model keeps learning until the gradients do not change or runs to the fixed number of epochs.

4 Experimental Results and Analysis

To evaluate the performance of the proposed method, we conducted two categories of experiments on several databases: one was carried out to extract features on standard MNIST database and AR face database, the other was carried out to classify on standard MNIST database, variation of MNIST database and OCR letters database. The experimental results showed that the reRBM could be more effective than RBM.

4.1 Features Extracted by ReRBM

We verified the efficiency of the reRBM using standard MNIST database and AR face database. Standard MNIST database contains 28×28 images which contains a training set with 60000 examples and a test set with 10000 examples, each image is handwritten digit number from 0 to 9 with white character on a

black background. AR face database contains 100 people's faces which contains a training set with 700 images and a test set with 699 images, each image is of 60 by 43 pixels with different facial expressions, illumination conditions. In this part, we only compare the features of input data extracted by reRBM and RBM, the experiments were performed on the training sets of two databases.

Comparison of Features on Standard MNIST Database. We compared the efficacy of reRBM and RBM on standard MNIST database. For fair comparison, the parameters of the two models were the same. We set initial values of bias to zero and set weight matrices to random values from uniform distribution $[-b^{-0.5}, b^{-0.5}]$, where b is the maximum value between the numbers of rows and columns of the matrix, and set learning rate to 0.005. For a better illustration of the features extracted by reRBM, we carried out the experiments using reRBM and RBM with different number of hidden neurons. Because the initial values of the weight matrices were random and the values of visible and hidden neurons are sampled, the experimental results were processed 10 times to ensure the effectiveness.

Figure 2 shows reconstructions for five examples. The results in row 1 were generated by two models with 64 hidden neurons, the results in row 2 were generated by the models with 128 hidden neurons, and the last row were generated by the models with 256 hidden neurons. The left displays the reconstructions generated by the RBM, and the right is the results produced by the reRBM. From Fig. 2, we could find that the reconstructions of the reRBM are better than those of RBM, especially, the reconstructions in row 1 generated by the reRBM. But as the number of hidden neurons increases, the difference between the two models is getting smaller and smaller. In short, it indicates that the reRBM obtains a better performance on extracting features compared to the RBM.

Fig. 2. Reconstructions for five examples from standard MNIST database (The left generated by the RBM, while the right generated by the reRBM).

Figure 3 shows the energy of two models with 1024 hidden neurons, the mean and standard deviation of reconstruction errors of two models with 6000 hidden neurons. In order to illustrate the difference between the two models, the figure only shows the result of the first 20 epochs. From Fig. 3, we can see that the convergence rate of the reRBM is faster than that of the RBM, and the reRBM could be more competitive with the RBM when the number of hidden neurons was small.

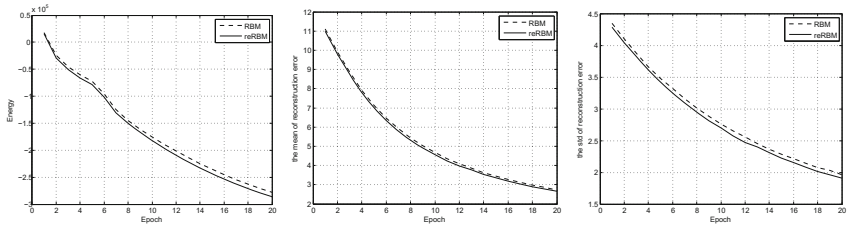

Fig. 3. Two models on standard MNIST. (left) the energies. (middle) the mean of reconstruction errors. (right) the standard derivation of reconstruction errors.

Comparison of Features on AR Face Database. We compared the performance of reRBM and RBM on AR face database. Because the face database is continuous data, the visible neurons of models are replaced by Gaussian units, and the hidden neurons remain binary. The value of visible neuron m is to sample from a normal distribution with unit variance and mean $a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn} h_n$, and the reconstruction data $\mathbf{v}'$ is defined as $v'_m = a_m + \sum_{n=1}^{|\mathbf{h}|} W_{mn} h_n$.

The network is trained using the gradient ascent method (the learning rate was set to 0.001, the weight matrices were initialized to $W_{mn} \sim 0.1 \times N(0, 1)$, and all initial values of biases were set to zero). Similarly, we performed the experiments with different number of hidden neurons and carried out ten times to ensure the effectiveness of the experimental results.

Figure 4 shows the reconstruction data for three examples. The faces in row 1 are generated by two models with 256 hidden neurons, the faces in row 2 are generated by the models with 1024 hidden neurons, and the last row are generated by the models with 3000 hidden neurons. We could conclude that the reconstruction results of the reRBM with Gaussian units are better than those of RBM with Gaussian units from Fig. 4.

Fig. 4. Reconstructions for three examples from AR face database (The left generated by the RBM with Gaussian units, while the right generated by the reRBM with Gaussian units).

Figure 5 shows the energy of two models with 1024 hidden neurons, the mean and standard deviation of reconstruction errors of two models with 3000 hidden neurons. From Fig. 5, we can conclude that the reRBM with Gaussian units obtains better performance on extracting features compared to the RBM with Gaussian units.

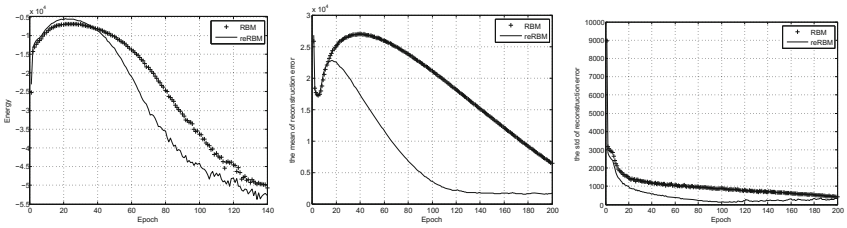

Fig. 5. Two models on AR. (left) the energies. (middle) the mean of reconstruction errors. (right) the standard derivation of reconstruction errors.

4.2 Classification Performance of ReRBM

For a classification problem, a label layer should be added on the RBM, and matrix **U** denotes the connections among the label layer and hidden layer. In the classification results, we focused on whether the reRBM could outperform the RBM.

We verify the classification performance of the proposed method using standard MNIST database, variation of MNIST database and OCR letters database. Each image of variation of MNIST database is handwritten digit number from 0 to 9 with black character on a white background, and the rest is the same as standard MNIST database. OCR letters database contains images of handwritten letters from a to z. All training sets were divided into two parts: one part is used for training and the other is used to validate. In this part, the number of validation part of the three databases was set to $10,000$, and the remaining part of the training set is used for training.

The parameters of the two models were the same as those in the experimental setting used by Larochelle [6]. The results of the experiments are shown in Table 1. Owing to the random initial values of the matrices and random sampling, the trials were executed 10 times. The experimental result for the RBM on standard MNIST database is 3.39% [6], the classification error rate of the RBM on variation MNIST is 3.16% [15]. The rest values in Table 1 are given by the mean of ten results. Table 1 shows that the classification results of reRBM are better than those of RBM.

Table 1. Classification error rates for three databases.

	RBM	reRBM
Standard MNIST	3.39%	**2.52**%
Variation MNIST	3.16%	**2.68**%
OCR	15.33%	**13.36**%

5 Conclusions

The RBMs have already been successfully applied to many tasks. Usually, the objective of the RBMs is maximizing the log-likelihood to make the distribution learned by the RBM as identical as the distribution of the input data. But reconstruction error always exists even the distribution learned by the RBM is identical as that of the input data. In this paper, a method to improve the performance of the RBM by adding the reconstruction error to the cost function of RBM was proposed. Two categories of experiments on several databases were carried out, the experimental results on standard MNIST and AR showed that reconstruction performance of the reRBM was better than that of the RBM, and classification results on standard MNIST, variation of MNIST and OCR letters showed that the reRBM was more competitive than the RBM. In future work, we intent to use the proposed method to more databases or other applications and apply the idea to other models.

Acknowledgments. This work was Supported by National Natural Science Fund for Distinguished Young Scholar Grant No. 61625204).

References

1. Fischer, A., Igel, C.: Training restricted Boltzmann machines: an introduction. Pattern Recogn. **47**(1), 25–39 (2014)
2. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted Boltzmann machines for collaborative filtering. In: International Conference on Machine Learning, vol. 227, pp. 791–798. ACM (2007)
3. Xie, G.S., Zhang, X.Y., Zhang, Y.M., Liu, C.L.: Integrating supervised subspace criteria with restricted Boltzmann machine for feature extraction. In: International Joint Conference on Neural Networks, pp. 1622–1629. IEEE Press, New York (2014)
4. Zhang, K., Liu, J., Chai, Y., Qian, K.: An optimized dimensionality reduction model for high-dimensional data based on restricted Boltzmann machines. In: Control and Decision Conference, pp. 2939–2944. IEEE Press, New York (2015)
5. Salakhutdinov, R., Tenenbaum, J.B., Torralba, A.: Learning with hierarchical-deep models. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1958–1971 (2013)
6. Larochelle, H., Mandel, M., Pascanu, R., Bengio, Y.: Learning algorithms for the classification restricted Boltzmann machine. J. Mach. Learn. Res. **13**(1), 643–669 (2012)

7. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
8. Hinton, G.E.: To recognize shapes, first learn to generate images. Prog. Brain Res. **165**, 535–547 (2007)
9. Ji, N.N., Zhang, J.S., Zhang, C.X.: A sparse-response deep belief network based on rate distortion theory. Pattern Recogn. **47**(9), 3179–3191 (2014)
10. Srivastava, N., Salakhutdinov, R.: Multimodal learning with deep Boltzmann machines. In: International Conference on Neural Information Processing Systems, pp. 2222–2230. IEEE Press, New York (2012)
11. Luo, L., Wang, Y., Peng, H., Tang, Z., You, S., Huang, X.: Training restricted Boltzmann machine with dynamic learning rate. In: International Conference on Computer Science & Education. IEEE (2016)
12. Yu, J., Gwak, J., Lee, S., Jeon, M.: An incremental learning approach for restricted Boltzmann machines. In: International Conference on Control, Automation and Information Sciences, pp. 113–117. IEEE Press, New York (2015)
13. Golovko, V., Kroshchanka, A., Turchenko, V., Jankowski, S., Treadwell, D.: A new technique for restricted Boltzmann machine learning. In: International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, vol. 1, pp. 182–186. IEEE Press, New York (2015)
14. Huang, W., Hong, H., Bian, K., Zhou, X., Song, G., Xie, K.: Improving deep neural network ensembles using reconstruction error. In: 2015 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. IEEE Press, New York (2015)
15. Yin, J., Lv, J., Sang, Y., Guo, J.: Classification model of restricted Boltzmann machine based on reconstruction error. Neural Comput. Appl. 1–16 (2016)
16. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)