2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Restricted Boltzmann Machine Associative Memory

Koki Nagatani and Masafumi Hagiwara

***Abstract*— Restricted Boltzmann machine associative memory (RBMAM) is proposed in this paper. RBMAM memorizes patterns using contrastive divergence learning procedure. It recalls by calculating the reconstruction of pattern using conditional probability. In order to examine the performance of the proposed RBMAM, extensive computer simulations have been carried out. As the result, it has shown that the performance of RBMAM is overwhelming compared with the conventional neural network associative memories. For example as for storage capacity, RBMAM can store about from $2N_{hidden}$ to $4N_{hideen}$ patterns, where N_{hidden} denotes the number of neurons in the hidden layer. Similarly we have obtained superior performance of RBMAM in respect of noise tolerance and pattern complement.**

I. Introduction

ASSOCIATIVE MEMORY is a kind of memory schemes what a part of memory content itself, not an address, is used as a key for recalling a memory content. An associative memory is considered to be the memory scheme of the brain [1], and has the following three features.

- The ability to correct faults if false information is given.
- To complete information if some parts are missing.
- To interpolate information, that means if a pattern is not stored, the most similar stored pattern is determined.

In the early 1970s, associative memory models implemented with neural networks were proposed almost simultaneously by three researchers, Nakano, Kohonen, and Anderson. Since then, a lot of neural associative memory models have been proposed and even now, neural associative memory models have been researched, improved, and applied as described hereinbelow. The earlier models are so-called Willshaw model [2] and Hopfield network models [3]. Hopfield network is a recurrent neural network having synaptic connection such that there is an underlying Lyapunov function for the activity dynamics. The phenomenon of associative memory matches the idea of dynamics controlled by a Lyapunov function.

Kobayashi recently proposed Hyperbolic Hopfield Neural Networks using Clifford algebra [4]. Kojima et al. [5] applied Boltzmann machine learning to an associative memory model and evaluated the capacity of an associative memory by numerical experiments in the case where the size of the network is small. Boltzmann machine is a type of stochastic recurrent neural network and can be seen as the stochastic, generative counterpart of Hopfield network. In [6], compared with the capacity of Hopfield Associative Memory, denoting the number of neurons as N, around $0.14N$, the capacity of the associative memory using Boltzmann machine is around $0.60N$. However better result this model is, learning is typically very slow in Boltzmann machine with many hidden layers because large networks should take a long time to approach their equilibrium distribution. Bayesian Confidence Propagation Neural Networks (BCPNNs) [7], which is a neural network implementing a naive Bayesian classifier, can be used as hetero-associative memories.

There is another associative memory model named Bidirectional Associative Memory [8]. BAM is primarily proposed by Kosko and many improvements, analyses and applications have been made [9]–[11]. BAM can associate one-to-one relationship bidirectionally and be used as a heteroassociative memory. Multidirectional Associative memory (MAM) [12] is a natural extension of BAM, which can have as many layers as needed. In [13], a competitive neural network is applied to MAM and more flexible layer settings could be done.

As the applications of associative memory models, in vision-related areas, Kuo et al. [14] applied it to detecting salient fragments for video human action detection and recognition. Arya et al. [15] recognized human faces using Parallel Associative Memory. Qadir et al. [16] [17] constructed and evaluated the performance of associative memories when they are used as a controller of a robot where sensory inputs contain errors. Wen et al. [18] established memristive neural network based on the knowledge of memristor and recurrent neural network and applied for adaptive lag synchronization. For intelligent systems, it is essential to adopt a neural associative memory with high performance.

Nowadays, deep learning has attracted much attention. Deep learning resembles the structure of the brain and is one method of multi-layer neural networks. As the famous models of deep learning method, autoencoder [19] and restricted Boltzmann machine (RBM) [20] [21] are proposed. They are often called representation learning models because of their learning method. For recognition tasks, these models are used as a single layer of the network and the pile of them compose a multi-layer neural network. An autoencoder is an artificial neural network used for learning efficient codings. If linear neurons are used, or only a single sigmoid hidden layer, then the optimal solution to an autoencoder is strongly related to PCA [22]. RBM is a probabilistic graphical model that can be interpreted as stochastic neural networks. The increase in computational power and the development of faster learning algorithms [23]–[25] have made them applicable to relevant machine learning problems. Le et al. [26] proposed large-scale feature detector using

Koki Nagatani and Masafumi Hagiwara are with the School of Science for Open and Environmental Systems, Graduate School of Science and Technology, Keio University, Hiyoshi, Kohoku-ku, Yokohama, Kanagawa Pref., Japan (email: {nagatani, hagiwara}@soft.ics.keio.ac.jp).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

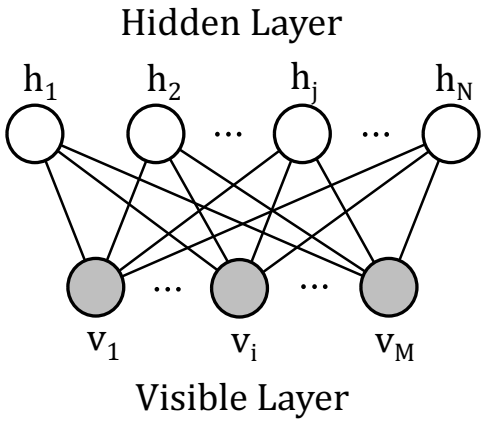


Fig. 1. The structure of RBM

unsupervised learning model for multi-layer neural network. It has been in the news that 1,000 computers were connected, trained by 10 million Youtube movies, and they learned to identify cat faces without human's help. In automatic speech recognition (ASR) systems, Gaussian mixture model hidden Markov model (GMM-HMM) used to be a main method. Recently, ASR system using deep neural network hidden Markov model hybrid architecture (DNN-HMM) is proposed and it can significant outperform the former model [27]–[29].

In this paper, we propose a new neural network associative memory named restricted Boltzmann machine associative memory (RBMAM). The vital learning method of deep learning is Greedy Layer-wise Training [30], which was proposed by Hinton et al. This method adjusts parameters layer by layer, making it approaching the input that inverse transforms the transformed input. Assuming that an input layer is presentation layer and higher layers than an input layer are internal of the brain, deep learning model can be defined as an associative memory. As the first step, this paper treats 2 layer type. The proposed RBMAM is superior storage capacity to the conventional neural network associative memories. Moreover, RBMAM has abilities of complementation and noise correction abilities.

The rest of this paper is organized as follows: Section II describes RBM and RBMAM in detail. Section III shows the results of conducted experiments to evaluate the proposed RBMAM. Finally, Section IV concludes the paper.

II. Restricted Boltzmann Machine Associative Memory

A. An overview of RBM

An RBM is a Markov random field (MRF) associated with a bipartite undirected graph as shown in Fig.1. It consists of M visible units $\boldsymbol{V} = (V_1, \cdots, V_M)$ to represent observable data and N hidden units $\boldsymbol{H} = (H_1, \cdots, H_N)$ to capture dependencies between observed variables. A joint configuration, $(\boldsymbol{V}, \boldsymbol{H})$ of the visible and hidden units has an energy given by:

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_i \theta_i^v v_i - \sum_j \theta_j^h h_j - \sum_i \sum_j v_i W_{ij} h_j, \tag{1}$$

where v_i and h_j are the binary states of visible unit i and hidden unit j, θ_i^v and θ_j^h are their biases and $\boldsymbol{W}$ is the weight of them.

The network assigns a probability to every possible pair of a visible and a hidden vector via this energy function:

$$p(\boldsymbol{v}, \boldsymbol{h}) = \frac{1}{Z} \exp(-E(\boldsymbol{v}, \boldsymbol{h})) \tag{2}$$

where the partition function, Z, is given by summing over all possible pairs of visible and hidden vectors:

$$Z = \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \exp(-E(\boldsymbol{v}, \boldsymbol{h})). \tag{3}$$

RBM trains using the probability that the network assigns to a visible vector $\boldsymbol{v}$, which is given by summing over all possible hidden vectors:

$$p(\boldsymbol{v}) = \sum_{\boldsymbol{h}} p(\boldsymbol{v}, \boldsymbol{h}). \tag{4}$$

The probability that the network assigns to a training dataset can be raised by adjusting the weights and biases to lower the energy of data of the dataset and to raise the energy of other data. Maximum log-likelihood estimation is often used to train. The log probability of a training vector is defined:

$$\begin{aligned} \mathcal{L} &\equiv \left\langle \ln \sum_{\boldsymbol{h}} p(\boldsymbol{v}, \boldsymbol{h}) \right\rangle_q \\ &= \left\langle \ln \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v}, \boldsymbol{h})} \right\rangle_q - \ln Z \end{aligned} \tag{5}$$

where the angle brackets are used to denote expectations under the distribution specified by the subscript, the probability distribution $q(\boldsymbol{v})$:

$$\left\langle f(\boldsymbol{v}) \right\rangle_q = \sum_{\boldsymbol{v}} f(\boldsymbol{v}) q(\boldsymbol{v}). \tag{6}$$

The derivative of the log probability of a training vector with respect to a voluntary parameter θ is calculated:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \theta} &= -\Big\langle \frac{1}{\sum_{\boldsymbol{h}}} \sum_{\boldsymbol{h}} e^{-E} \frac{\partial E}{\partial \theta} e^{-E} \Big\rangle_q \\ &\quad + \frac{1}{Z} \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \frac{\partial E}{\partial \theta} e^{-E} \end{aligned} \tag{7}$$

and according to the definition of conditional probability and MRF, Eq. (7) can be simplified as follows:

$$\frac{\partial \mathcal{L}}{\partial \theta} = \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \frac{\partial E}{\partial \theta} p(\boldsymbol{h}|\boldsymbol{v}) q(\boldsymbol{v}) + \sum_{\boldsymbol{v}} \sum_{\boldsymbol{h}} \frac{\partial E}{\partial \theta} p(\boldsymbol{v}, \boldsymbol{h}) \tag{8}$$

$$\equiv -\Big\langle \frac{\partial E}{\partial \theta} \Big\rangle_{data} + \Big\langle \frac{\partial E}{\partial \theta} \Big\rangle_{model}. \tag{9}$$

The graph of an RBM has only connections between the layer of hidden and visible variables but not between two variables of the same layer. In terms of probability this means that the hidden units are independent given the state of the visible units and vice versa:

$$p(\boldsymbol{v}|\boldsymbol{h}) = \prod_i p(v_i|\boldsymbol{h}, p(\boldsymbol{h}|\boldsymbol{v})) = \prod_j p(h_j|\boldsymbol{v}). \tag{10}$$

The conditional probability of a single variable being one can be interpreted as the firing rate of a neuron:

$$p(v_i = 1|\boldsymbol{h}) = \sigma(\theta_i^v + \sum_j W_{ij} h_j) \tag{11}$$

$$p(h_j = 1|\boldsymbol{v}) = \sigma(\theta_j^h + \sum_i W_{ij} v_i) \tag{12}$$

where σ is the sigmoid activation function:

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \tag{13}$$

B. Contrastive Divergence

In Eq. (9), the first member of the equation is easy to calculate comparatively. The second member of the equation, however, is much more difficult. This is because in general, this member needs to be calculated regarding the sum of all states of both visible units and hidden units. It causes the curse of dimensionality.

The standard approach to solve this problem is to approximate the average over the distribution with an average over a sample from the model distribution, obtained by setting up a Markov chain that converges to the model distribution and running the chain to equilibrium. This Markov chain Monte Carlo (MCMC) approach has the advantage of being readily applicable to many classes of distribution. However, it is typically very slow, since running the Markov chain to equilibrium can require a very large number of steps, and no foolproof method exists to determine whether equilibrium has been reached. A further disadvantage is the large variance of the estimated gradient [25].

Recently it was shown that estimates obtained after running the chain for just a few steps can be sufficient for model training. This leads to contrastive divergence (CD) learning [25], which has become a standard way to train RBM.

The idea of k-step contrastive divergence learning is quite simple: Instead of approximating the second term in the log-likelihood gradient by a sample from the RBM-distribution (which would require to run a Markov chain until the stationary distribution is reached), a Gibbs chain is running for only k steps (and usually $k = 1$). The Gibbs chain is initialized with a training example $\boldsymbol{v}^{(0)}$ of the training set and yields the sample $\boldsymbol{v}^{(k)}$ after k steps. Each step t consists of sampling $\boldsymbol{h}^{(t)}$ from $p(\boldsymbol{h}|\boldsymbol{v}^{(t)})$ and sampling $\boldsymbol{v}^{(t+1)}$ from $p(\boldsymbol{v}|\boldsymbol{h}^{(t)})$ subsequently. The gradient (Eq. (9)) with respect to the log-likelihood for one training pattern $\boldsymbol{v}^{(0)}$ is then approximated by

$$\begin{aligned} \mathrm{CD}_k(\theta, \boldsymbol{v}^{(0)}) = &- \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}^{(0)}) \frac{\partial E(\boldsymbol{v}^{(0)}, \boldsymbol{h})}{\partial \theta} \\ &+ \sum_{\boldsymbol{h}} p(\boldsymbol{h}|\boldsymbol{v}^{(k)}) \frac{\partial E(\boldsymbol{v}^{(k)}, \boldsymbol{h})}{\partial \theta}. \end{aligned} \tag{14}$$

The derivatives in direction of the single parameters are obtained by estimating the expectations over $p(\boldsymbol{v})$ by the single sample $\boldsymbol{v}^{(k)}$. The update equations of parameters, biases of visible and hidden units and weight of them, are calculated:

$$\begin{aligned} \Delta w_{ij} \leftarrow\ & \Delta w_{ij} + p(H_i = 1|\boldsymbol{v}^{(0)}) \cdot v_j^{(0)} \\ & - p(H_i = 1|\boldsymbol{v}^{(k)}) \cdot v_j^{(k)}, \end{aligned} \tag{15}$$

$$\Delta \theta_i^v \leftarrow \Delta \theta_i^v + v_i^{(0)} - v_i^{(k)}, \tag{16}$$

and

$$\Delta \theta_j^h \leftarrow \Delta \theta_j^h + p(H_i = 1|\boldsymbol{v}^{(0)}) - p(H_i = 1|\boldsymbol{v}^{(k)}). \tag{17}$$

C. Restricted Boltzmann machine associative memory

In this section, we explain the proposed restricted Boltzmann machine associative memory.

Training of RBM is making it approaching the input that inverse transforms the transformed input using its conditional probability. Although RBMs are often used to reduce the dimension, assuming that visible layer is a presentation layer and hidden layer is internal of the brain, RBMs can be considered as an associative memory. Therefore, we call the associative memory model using RBMs restricted Boltzmann machine associative memory (RBMAM).

Learning patterns of RBMAM is done by the same method of training of RBM. Although there are some learning methods of training RBMs [31]–[33], we use CD learning method for simplicity.

A recall is a process that if an unknown input pattern is given, the network selects the pattern of training patterns. It is identical to select the pattern which has the minimum Hamming distance to an unknown input pattern. The recall of RBMAM realizes to find the pattern $\boldsymbol{u}^*$ of training patterns by calculating:

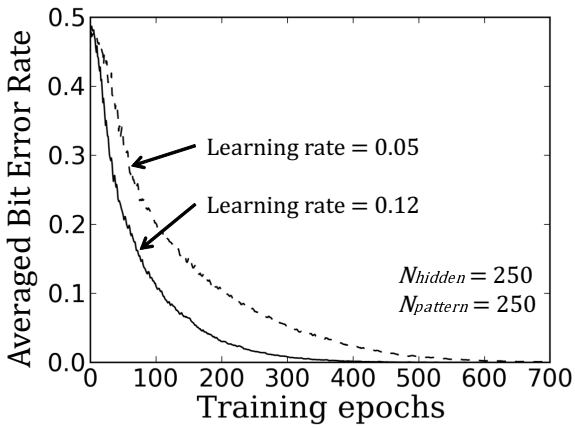
$$\boldsymbol{u}^* = \sigma(\sigma(\tilde{\boldsymbol{u}} \cdot \boldsymbol{W} + \boldsymbol{\theta}^h) \cdot \boldsymbol{W}^T + \boldsymbol{\theta}^v) \tag{18}$$

where $\boldsymbol{W}^T$ is the transpose of weight matrix $\boldsymbol{W}$ and σ is the sigmoid activation function (Eq. (13))

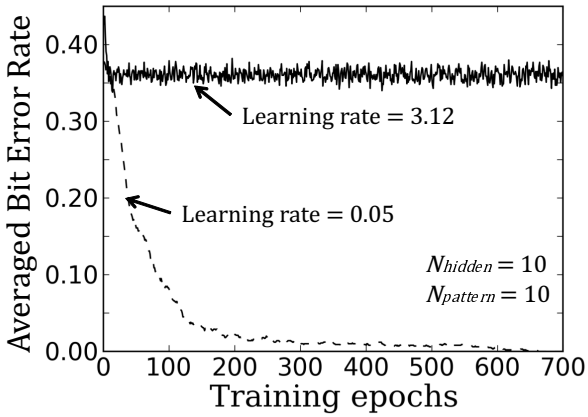
III. Experiments

In this section, we evaluate performance of RBMAM for average bit error rate, which means the ratio of the number of bit errors in all training patterns' bits.

Training patterns of RBMAM depends to a great degree on two parameters; the number of learning epochs and learning rate. There are some values or techniques to train fast, we referred to the G. E. Hinton's guide [34] about them. About these two parameters, we determined them by preliminary experiments and the result is shown in Fig. 2. Here, N_{hidden}



(a) Learning rate calculated by Hinton's guide is lower than 0.20.



(b) Learning rate calculated by Hinton's guide is higher than 0.20

Fig. 2. Average bit error rate as a function of the number of training epochs.

is the number of hidden neurons and $N_{pattern}$ is that of the input patterns. Concerning the number of training epochs, average bit error rate basically converges about 1,000 epochs, so we determine the number of training epoch 1,000 in the experiments. As to learning rate, it is appropriate to adjust so that the updates are about 10^{-3} times the weights in Hinton's guide [34]. However, learning rate depends on datasets and regarding small datasets if learning rate was more than about 0.2 when we initialized, learning became too slow and sometimes did not converge. In our pre-experiments, when learning rate was initialized at 0.05, learning converged in almost all datasets. Therefore, we determine learning rate as follows: if the initial learning rate is more than 0.2, it is revised to 0.05 and if less than 0.2, not revised and use the same value. For the other parameters, we used 100-dimensional, two-valued ($\{0, 1\}$) vectors for input. All of the data shown afterward are the averaged value of 10 trials.

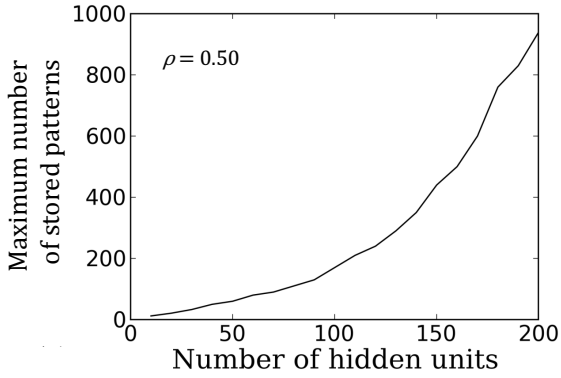


Fig. 3. Maximum number of stored patterns as a function of the number of hidden units. In the experiment 100-dimensional, binary vectors are used for input.

A. Storage Capacity (perfect recall)

The storage capacity is one of the most important properties of associative memories. To examine RBMAM's storage capacity, first we use the same pattern vectors for the training sets and the test sets.

Fig. 3 shows the maximum number of stored patterns as a function of the number of hidden units. In this case, RBMAM can perfectly recall the stored patterns. As can be seen from Fig. 3, when the number of hidden units is small, e.g., less than 100, the storage capacity of RBMAM is not large. However, when the number of hidden units becomes larger, RBMAM can recall about four times of the number of hidden units. It is well known that the storage capacity of Hopfield network is about $0.14N$ where N is the number of the neurons [35]. In contrast, RBMAM's storage capacity is, denoting the number of hidden units as N_{hidden}, about from twice to four times as many patterns as N_{hidden}. Therefore, RBMAM has a superior memory capacity to the conventional associative memories.

B. Influence of Cross-Correlation

It is well known that the storage capacity of an associative memory is affected depending on the cross-correlation of the stored patterns. We examined the storage capacities with several datasets having different cross-correlations. Cross-correlation is defined as

$$\rho = \frac{1}{{}_p\mathrm{C}_2} \sum_{\mu_1=1}^{M} \sum_{\mu_2=\mu_1+1}^{M} \frac{1}{2} \frac{|\boldsymbol{u}^{\mu_1 T} \boldsymbol{u}^{\mu_2}|}{M}. \tag{19}$$

Fig. 4 shows averaged bit error rate as a function of learning epochs when the averaged cross-correlation of 500 pattern vectors are 0.22, 0.50, and 0.89. In the experiment, the number of hidden units was 250, the number of training patterns was 500, and the same pattern vectors for training sets and test sets were used. As can be seen from Fig. 4, the averaged bit error rate approaches zero faster when the smaller value of cross-correlation is used.

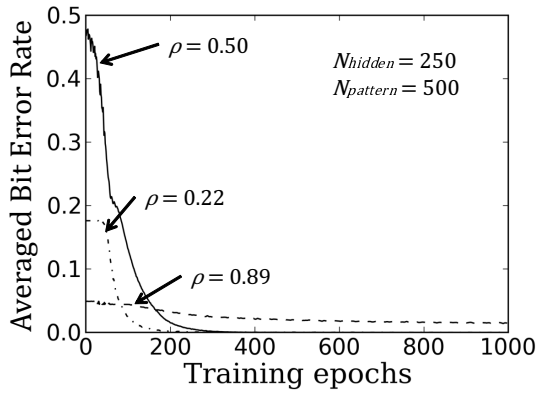


Fig. 4. Average bit error rate as a function of learning epochs when the cross-correlation of 500 pattern vector is for each 0.22, 0.50. and 0.89.

TABLE I

THE STORAGE CAPACITY OF RBMAMS WITH 250 HIDDEN UNITS. THE NOTATION $> n$ MEANS THAT THE 90% CAPACITY IS MORE THAN n

Cross-correlation ρ	Storage Capacity
0.22	$> 1,000$
0.50	$> 1,000$
0.89	290

Table I shows the storage capacity of RBMAM with 250 hidden units, which is defined as the maximum number of stored patterns where at least 90% of all stored patterns can be perfectly recalled in the experiment.

Namely more than 89 bits are correctly recalled out of 100 bits in this case. It can be seen that the smaller the cross-correlation of training patterns becomes, the higher storage capacity of RBMAMs becomes.

C. Noise Tolerance

We examined RBMAM's ability for noise tolerance.

Fig. 5 shows averaged bit error rate as a function of the number of training patterns when bits are randomly inverted with the probabilities from 0.1 to 0.5. It can be observed from this figure that the noise tolerance ability of RBMAM is superior. For example, when the number of input patterns is 500 and bit reverse probability is 10%, the proposed RBMAM can recall more than 90% of the stored patterns.

D. Pattern Complement

Finally, the pattern complenet ability of RBMAM was examined.

Fig. 6 shows averaged bit error rate as a function of the number of training patterns when bits are randomly blind. From bottom to top, the probabilities of bit change of these curbs are 10%, 20%, 30%, 40%, and 50%. We should note ere that since binary expression $\{0, 1\}$ is employed in the input vector, the elements having the value of "1" were randomly selected with several probabilities and their values were changed to "0".

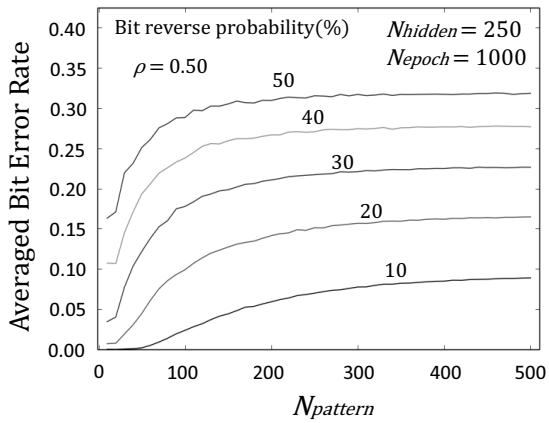


Fig. 5. Averaged bit error rate as a function of the number of training patterns. Test sets include from 10% to 50% bit invert. From bottom to top, the probabilities of bit change of these curbs are 10%, 20%, 30%, 40%, and 50%.

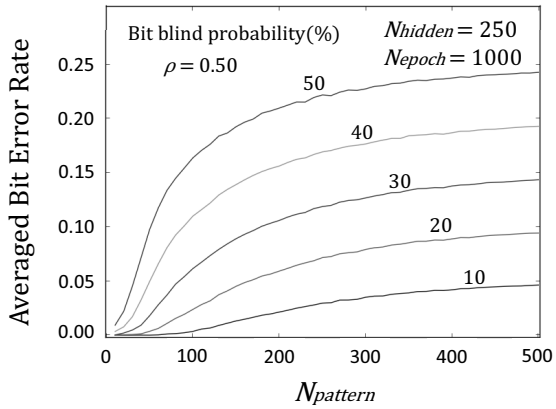


Fig. 6. Averaged bit error rate as a function of the number of training patterns. Test sets include from 10% to 50% bit blind. From bottom to top, the probabilities of bit change of these curbs are 10%, 20%, 30%, 40%, and 50%.

Similar to the case of noise tolerance ability, pattern complement ability of RBMAM is also superior. For example, when the number of input patterns is 500 and bit reverse probabilities is 20%, the proposed RBMAM can recall more than 90% of the stored patterns.

IV. CONCLUSIONS

In this paper, we have proposed a new kind of associative memory named Restricted Boltzmann Machine Associative Memory(RBMAM). RBMAM has the same network structure as a two-layer RBM. Contrastive divergence learning is employed as the learning algorithm. A recall is done by using conditional probability.

We evaluated the proposed RBMAM from various aspects such as memory capacity, noise tolerance and pattern complement. As the result, it has shown that the performance of

RBMAM is overwhelming compared with the conventional neural network associative memories. As for storage capacity, it can store about from $2N_{hidden}$ to $4N_{hidden}$ patterns, where N_{hidden} denotes the number of neurons in the hidden layer. As for noise tolerance, when bit reverse probability is 10%, it could recall more than 90% of the stored patterns. And as for pattern complement, when bit blind probability is 20%, the proposed RBMAM could recall more than 90% of the stored patterns.

REFERENCES

[1] A.Lanser,"Associative memory models: from the cell-assembly theory to biophysically detailed cortex simulations," *Trends in Neurosciences,* Vol. 32, No.3, pp. 178-186, March 2009.

[2] D. J. Willshaw, O. P. Buneman, and H. C. Longuet-Higgins, "Non-holographic associative memory," *Nature,* Vol. 222, No. 5197, pp.960-962, Jun. 1969.

[3] J.J. Hopfield, " Neural networks and physical systems with emergent collective computational abilities, " *Proceedings of the National Academy of Sciences of the United States of America,* Vol. 79, pp.2554-2558, April 1982.

[4] M. Kobayashi, "Hyperbolic Hopfield Neural Networks," *IEEE transactions on Neural Networks and Learning Systems*, Vol. 24, pp. 335-341, Feb. 2013.

[5] T.Kojima, H. Nagaoka, T.Da-Te, "Some properties of an associative memory model using the Boltzmann Machine learning," *Proc. the IEEE International Joint Conference on Neural Networks,* Vol. 3, pp.2662-2665, 1993.

[6] T.Kojima, H. Nonaka, T. Da-Te, "Capacity of the associative memory using the Boltzmann machine learning," *Proc. the IEEE International Conference on Neural Networks,* Vol. 5, pp.2572-2577, 1995.

[7] A. Lansner and O. Ekeberg, "A one-layer feedback artificial neural network with a Bayesian learning rule," *International Journal of Neural Systems,* Vol. 1, No. 1, pp.77-87, 1989.

[8] B. Kosko, " Bidirectional associative memories, " *IEEE Trans. on Systems, Man, and Cybernetics,* Vol.18, no.1, pp.49-60, 1988.

[9] M.Hattori and M.Hagiwara, "Intersection learning for bidirectional associative memory," *Proc. the IEEE International Conference on Neural Networks,* Vol. 1, pp.555-560, 1996.

[10] F. T. Sommer and G. Palm, "Improved bidirectional retrieval of sparse patterns stored by hebbian learning," *Neural Networks,* Vol.12, No. 2, pp.281-297, Mar. 1999.

[11] M. Wang and S. Chen, "An improvement of BAM in storage capacity and error-correction capability," *Proceedings of the 1st International Symposium on Data, Privacy, and E-Commerce,* pp.164-166, Nov. 2007.

[12] M. Hagiwara, "Multidirectional associative memory," *Proceedings of the International Joint Conference on Neural Networks,* pp.3-6, IEEE, 1990.

[13] H. Yu, F. Shen, O. Hasegawa, "A multidirectional associative memory based on self-organizing incremental neural network," *Proceedings of the International Conference on Neural Information Processing: Models and Applications,* Vol. 6444 of Lecture Notes in Computer Science, pp.344-351, Berlin, Heidelberg, 2010. Springer.

[14] Da-Wei Kuo, Guan-Yu Cheng, Shyi-Chyi Cheng and Su-Ling Lee, "Detecting salient fragments for video human action detection and recognition using an associative memory," *International Symposium on Communications and Information Technologies,* pp. 1039-1044, 2012.

[15] K. V. Arya, V. Singh, P. Mitra, and P. Gupta, "Face recognition using Parallel Associative Memory," *International Conference on Systems, Man and Cybernetics,* pp. 1332-1336, 2008.

[16] O. Qadir, J. Liu, J. Timmis, G. Tempesti and A. Tyrrell, "Principles of protein processing for a self-organizing associative memory," *IEEE Congress on Evolutionary Computation,* pp. 1-8. IEEE, July. 2010.

[17] O. Qadir, J. Liu, G.Tempesti, J. Timmis and A. Tyrrell, "From bidirectional associative memory to a noise-tolerant,robust protein processor associative memory," *Artificial Intelligence,* Vol. 175, No. 2, pp. 673-693, Feb. 2011.

[18] S. Wen, Z. Zeng, T. Huang, Y. Zhang, "Exponential Adaptive Lag Synchronization of Memristive Neural Networks via Fuzzy Method and Applications in Pseudo Random Number Generators," *IEEE Transactions on Fuzzy Systems,* DOI: 10.1109/TFUZZ.2013.2294855, Dec. 2013.

[19] Y. Bengio, "Learning Deep Architecture for AI," *Foundations and Trends in Machine Learning,* Vol. 2, pp.1-15, 2009.

[20] G.E.Hinton and R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science 313(5786),* pp. 504-507, 2006

[21] P.Smolensky, "Chapter 6: Information Processing in Dynamical Systems: Foundations of Harmony Theory". In Rumelhart, David E.; McLelland, James L. Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations. MIT Press. pp. 194-281.

[22] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biological Cybernetics,* Vol. 59, pp.291-294, 1988

[23] P. Vincent, H. Larochelle, Y. Bengio, and P. Manzagol., "Extracting and composing robust features with denoising autoencoders." *International Conference on Machine Learning,* 2008.

[24] G.E.Hinton, N.Srivastava, A.Krizhevsky, I.Surskever, and R.R.Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *Neural and Evolutionary Computing,* 2012

[25] M.A.Carreira-Perpinan, G.E.Hinton, "On Contrastive Divergence Learning," *Artificial Intelligence and Statistics,* 2005

[26] Quoc Le, Marc'Aurelio Ranzato, Rajat Monga, Matthieu Devin, Kai Chen, Greg Corrado, Jeff Dean, Andrew Ng, "Building high-level features using large scale unsupervised learning," *International Conference in Machine Learning,* 2012

[27] George E. Dahl, Dong Yu, Li Deng, and Ales Acero, "Context-Dependant Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing,* Vol.20, No.1, Jan. 2012.

[28] E. Arisoy, T.N.Sainath, B.Kingsbury, and B. Ramabhadran, "Deep Neural Network Language Models," *The Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* pp.20-28, 2012

[29] A.Mnih and Y.W.Teh, "A fast and simple algorithm for training neural probabilistic language models," *International Conference on Machine Learning,* 2012

[30] Y.Bengio, P.Lamblin, D.Popovici, and H.Larochelle, "Greedy layer-wise training of deep networks," *Adv, in Neural Information Processing Systems,* 2007

[31] T. Tieleman, "Training Restricted Boltzmann Machine using Approximations to the Likelihood Gradient," *Proceedings of the 25th International Conference on Machine Learning,* pp.1064-1071, 2008.

[32] R. Salakhutdinov and I. Murray, "On the Quantitative Analysis of Deep Belief Networks," *Proceedings of the 25th international conference on Machine learning,* pp.872-879, 2008.

[33] KH. Cho, T. Raiko, and A. Ilin, "Parallel Tempering is Efficient for Learning Restricted Boltzmann Machines," *International Joint Conference on Neural Networks,* pp.1-8, IEEE, 2010.

[34] Geoffrey Hinton, "A Practical Guide to Training Restricted Boltzmann Machines version 1, " 2010.

[35] R. J. McEliece, E. C. Posner, E. R. Rodemich, and S. S. Venkatesh, "The capacity of the Hopfield associative memory," *IEEE Trans. Inform. Theory,* ol.33, pp.461-482, 1987.