SCIENTIFIC REPORTS

OPEN

Temperature based Restricted Boltzmann Machines

Guoqi Li[1,*], Lei Deng[1,*], Yi Xu[2,*], Changyun Wen[3], Wei Wang[4], Jing Pei[1] & Luping Shi[1]

Received: 27 July 2015
Accepted: 02 December 2015
Published: 13 January 2016

Restricted Boltzmann machines (RBMs), which apply graphical models to learning probability distribution over a set of inputs, have attracted much attention recently since being proposed as building blocks of multi-layer learning systems called deep belief networks (DBNs). Note that temperature is a key factor of the Boltzmann distribution that RBMs originate from. However, none of existing schemes have considered the impact of temperature in the graphical model of DBNs. In this work, we propose temperature based restricted Boltzmann machines (TRBMs) which reveals that temperature is an essential parameter controlling the selectivity of the firing neurons in the hidden layers. We theoretically prove that the effect of temperature can be adjusted by setting the parameter of the sharpness of the logistic function in the proposed TRBMs. The performance of RBMs can be improved by adjusting the temperature parameter of TRBMs. This work provides a comprehensive insights into the deep belief networks and deep learning architectures from a physical point of view.

A restricted Boltzmann machine (RBM) is a generative stochastic artificial neural network[1–6] that applies graphical models to learning a probability distribution over a set of inputs[7]. The restricted Boltzmann machines (RBMs) were initially invented under the name Harmonium by Smolensky in 1986[8]. After that, Hinton *et al.* proposed fast learning algorithms for training a RBM in mid-2000s[9–12]. Since then, RBMs have found wide applications in dimensionality reduction[9], classification[13–18], feature learning[19–25], pattern recognition[26–29], topic modelling[30] and various other applications[31–39]. Generally, RBMs can be trained in either supervised or unsupervised ways, depending on the task. RBMs originate from the concept of Boltzmann distribution[40], a well known concept in physical science where temperature is a key factor of the distribution. In fact, in statistical mechanics[41–43] and mathematics, a Boltzmann distribution is a probability distribution of particles in a system over various possible states. Particles in this context refer to gaseous atoms or molecules, and the system of particles is assumed to have reached thermodynamic equilibrium[44,45]. The distribution is expressed in the form of $F(state) \propto e^{-\frac{E}{kT}}$ where E is state energy which varies from state to state, and kT is the product of Boltzmann's constant and thermodynamic temperature. However, none of existing schemes in RBMs consider the temperature parameter in the graphical models, which limits the understanding of RBM from a physical point of view.

In this work, we revise the RBM by introducing a parameter T called "temperature parameter", and propose a model named "temperature based restricted Boltzmann machines" (TRBMs). Our motivation originates from the physical fact that the Boltzmann distribution depends on temperature while so far in RBM, the effect of temperature is not considered. The main idea is illustrated in Fig. 1. From a mathematical point of view, the newly introduced T is only a parameter that gives more flexibility (or more freedom) to the RBM. When $T = 1$, the model TRBM reduces to the existing RBM. So the present RBM is a special case of the TRBM. We further show that the temperature parameter T plays an essential role which controls the selectivity of the firing neurons in the hidden layers. In statistical mechanics, Maxwell-Boltzmann statistics[46–48] describes the average distribution of non-interacting material particles over various energy states in thermal equilibrium, and is applicable when the temperature is high enough or the particle density is low enough to render quantum effects negligible. Note that the change in temperature affects the Maxwell-Boltzmann distribution significantly, and the particle distribution depends on the temperature (T) of the system. At a lower temperature, distributions moves to the left side with a higher kurtosis[49]. This implies that a lower temperature leads to a lower particle activity but higher entropy[50–52]. In this paper, we uncover that T affects the firing neurons activity distribution similar to that of a temperature

[1]Center for Brain Inspired Computing Research, Department of Precision Instrument, Tsinghua University, Beijing, China, 100084. [2]School of Computing Enginering, Nanyang Technological University, Singapore, 639798. [3]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, 639798. [4]School of Automation Science and Electrical Engineering, Beihang University, Beijing, China, 100191. [*]These authors contributed equally to this work. Correspondence and requests for materials should be addressed to G.L. (email: liguoqi@mail.tsinghua.edu.cn) or L.S. (email: lpshi@mail.tsinghua.edu.cn)

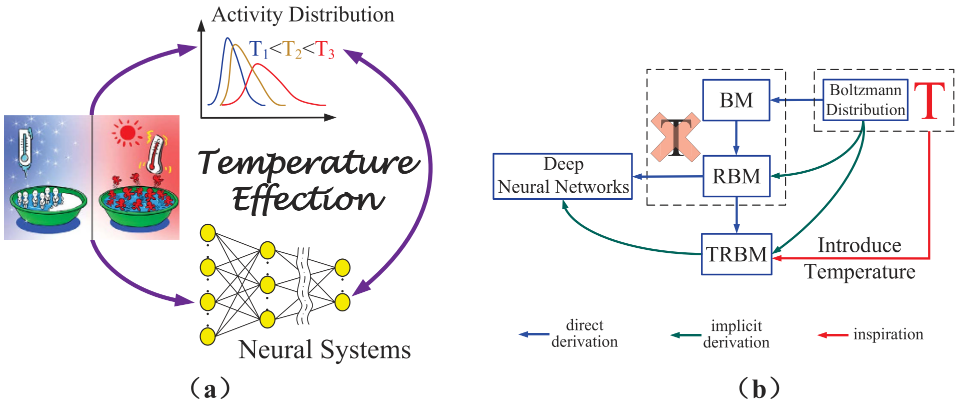

Figure 1. The main idea of this work. (**a**) The relationships among the temperature in a real-life physical systems, particle activity distribution and the artificial neural systems. (**b**) Illustration of a TRBM as a variant of Boltzmann machine (BM) and restricted Boltzmann machine (RBM).

parameter in Boltzmann distribution illustrated in Fig. 1, which gives some insights on the newly introduced *T* from physical point of view. From the figure, it is seen that a TRBM is a variant of the Boltzmann machine (BM) and RBM named after Boltzmann distribution. Note that the difference between BM and RBM lies in the various constraints on the connections between neurons, while the energy function in both BM and RBM follows the Boltzmann distribution. So far in both BM and RBM, the effect of temperature is not considered, especially when used for machine learning. In this work, we address such an issue by introducing a temperature parameter *T* into the probability distribution of the energy function following the Boltzmann distribution.

Our approaches and contributions are summarized as follows. Firstly, we prove that the effect of the temperature parameter can be transformed to the steepness of the logistic function[53,54] which is a common "S" shape (sigmoid curve) when employing the contrastive divergence algorithm in the process of pre-training of a TRBM. Because the steepness of the sigmoid curve changes the accept probability when employing the Markov Chain Monte Carlo (MCMC) methods[55,56] for sampling the Markov random process[57,58]. Secondly, it is proven that the error propagated from the output layer will be multiplied by $\frac{1}{T}$, i.e., the inverse of the temperature parameter *T*, in every layer when doing a modified back propagation (BP) in the process of fine-tuning of a TRBM. We also show that the propagated error further affects the selectivity of the features extracted by the hidden layers. Thirdly, we show that the neural activity distribution impacts the performance of the TRBMs. It is found that the relatively lower temperature enhances the selectivity of the extracted features, which improves the performance of a TRBM. However, if the temperature is lower than certain value, the selectivity turns to deteriorate, as more and more neurons become inactive. Based on the results established in this paper, it is natural to imagine that temperature may affect the cognition performance of a real neural system.

Results

Temperature based Restricted Boltzmann Machines.

As mentioned, a RBM is a generative stochastic artificial neural network that can learn a probability distribution over a given set of inputs. A RBM is a variant of the original Boltzmann machine, which requires all neurons to form a bipartite graph – neurons are divided into two groups, where one group contains "visible" neurons and the other group contains "hidden" ones. Neurons from different groups may have a symmetric connection, but there is no connection among neurons within the same group. This restriction allows for more efficient training algorithms which are available for the original class of Boltzmann machines, in particular the gradient-based contrastive divergence algorithm[59,60].

It is well known that a RBM is an energy-based model in which the energy function[61–63] is defined as

$$E_\theta(v,\, h) = \sum_{i=1}^{n_v} a_i v_i + \sum_{j=1}^{n_n} b_j v_j + \sum_{i=1}^{n_v}\sum_{j=1}^{n_h} h_j w_{i,j} v_i \tag{1}$$

where θ consists of $W = (w_{i,j})$ (size $n_v \times n_h$) which is associated with the connection between a hidden unit h_j and a visible unit v_i, and bias weights a_i for visible units and b_j for hidden units. To incorporate the temperature effect into the RBM, a temperature parameter *T* is introduced to the following joint distribution of the vectors *v* and *h* of the "visible" and "hidden" vectors:

$$P_\theta(v,\, h,\, T) = \frac{1}{Z_\theta(T)} e^{\frac{-E_\theta(v,h)}{T}} \tag{2}$$

where $Z_\theta(T)$, the sum of $e^{\frac{-E_\theta(v,h)}{T}}$ over all possible configurations, is a normalizing constant which ensures the probability distribution sums to 1, i.e.,

$$Z_\theta(T) = \sum_{v,h} e^{\frac{-E_\theta(v,h)}{T}} \tag{3}$$

Denote $\boldsymbol{v}$ as an observed vector of v. Similar to RBM, TRBM is trained to maximize a distribution called likelihood function $P_\theta(\boldsymbol{v})$, which is a marginal distribution function of $P_\theta(v, h, T)$, i.e.,

$$P_\theta(\boldsymbol{v}) = \sum_h P_\theta(\boldsymbol{v},\ h,\ T) = \frac{1}{Z_\theta(T)} \sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}} = \frac{1}{\sum_{v,h} e^{\frac{-E_\theta(v,h)}{T}}} \sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}} \tag{4}$$

It is obtained that

$$\ln(P_\theta(\boldsymbol{v})) = \ln\left(\sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}}\right) - \ln\left(\sum_{v,h} e^{\frac{-E_\theta(v,h)}{T}}\right) \tag{5}$$

Remark 1. Similarly to RBM, there are two stages for training a typical TRBM for deep learning, i.e., a pre-training stage and a fine-tuning stage[10–12]. The most frequently used algorithms for these two stages are contrastive divergence and back propagation, respectively.

Contrastive divergence for pre-training a TRBM.

In the pre-training stage, the contrastive divergence algorithm performs MCMC/Gibbs sampling and is used inside a gradient descent procedure to compute weight update. Theorem 1 shows that we only need to modify the sharpness of a logistic function when temperature is considered, in order to employ the contrastive divergence algorithm.

Theorem 1. When applying contrastive divergence for pre-training a TRBM, the temperature parameter controls the sharpness of the logistic sigmoid function.

Proof. Note that for an observed $\boldsymbol{v}$, we have

$$\frac{1}{\sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}}} e^{-\frac{E_\theta(\boldsymbol{v},h)}{T}} = \frac{\frac{e^{-\frac{E_\theta(\boldsymbol{v},h)}{T}}}{Z}}{\frac{\sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}}}{Z}} = \frac{P(\boldsymbol{v},\ h)}{P(\boldsymbol{v})} = P(h|v) \tag{6}$$

Then, the likelihood function can be written as

$$\begin{aligned} \frac{\partial\ \ln(P_\theta(\boldsymbol{v}))}{\partial\theta} &= \frac{\partial}{\partial\theta} \ln\left(\sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}}\right) - \frac{\partial}{\partial\theta}\ \ln\left(\sum_{v,h} e^{\frac{-E_\theta(v,h)}{T}}\right) \\ &= -\frac{1}{\sum_h e^{\frac{-E_\theta(\boldsymbol{v},h)}{T}}}\left[\sum_h e^{-\frac{E_\theta(\boldsymbol{v},h)}{T}} \frac{\partial E_\theta(\boldsymbol{v},\ h)}{\partial\theta}\right] \\ &\quad + \frac{1}{\sum_{v,h} e^{\frac{-E_\theta(v,h)}{T}}}\left[\sum_{v,h} e^{-\frac{E_\theta(v,h)}{T}} \frac{\partial E_\theta(v,\ h)}{\partial\theta}\right] \\ &= -\sum_h P(h|\boldsymbol{v}) \frac{\partial E_\theta(\boldsymbol{v},\ h)}{\partial\theta} + \sum_{v,h} P(v,\ h) \frac{\partial E_\theta(v,\ h)}{\partial\theta} \\ &= -\sum_h P(h|\boldsymbol{v}) \frac{\partial E_\theta(\boldsymbol{v},\ h)}{\partial\theta} + \sum_v P(v) \sum_h P(h|v) \frac{\partial E_\theta(v,\ h)}{\partial\theta} \end{aligned} \tag{7}$$

Denote that

$$\begin{aligned} h_{-i} &= \left[h_1 h_2 \ldots h_{k-1} h_{k+1} \ldots h_{n_h}\right]^T \\ \alpha_k(v) &= b_k + \sum_{i=1}^{n_v} w_{k,i} v_i \\ \beta(v,\ h_{-i}) &= \sum_{i=1}^{n_v} a_i v_i + \sum_{j=1, j\neq k}^{n_h} b_j h_j + \sum_{i=1}^{n_v} \sum_{j=1, j\neq k} h_j w_{j,i} v_i \\ E_\theta(v,\ h) &= -\beta(v,\ h_{-i}) - h_k \alpha_k(v) \end{aligned} \tag{8}$$

When $\theta = W_{ij}$, we have

$$\begin{aligned}
\sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(v,\,h)}{\partial W_{ij}} &= \sum_h P(h|\boldsymbol{v})h_i v_j \\
&= \sum_h \prod\nolimits_{k=1}^{n_h} P(h_k|\boldsymbol{v})h_i v_j \\
&= \sum_{h_j} P(h_i|\boldsymbol{v})h_i v_j \\
&= -\Big(P(h_i = 0|v)\cdot 0\cdot v_j + P(h_i = 1|v)\cdot 1\cdot v_j\Big) \\
&= -\,P(h_i = 1|v)v_j \\
&= -\,P(h_i = 1|h_{-i},\,v)v_j \\
&= -\,\frac{P(h_i = 1,\,h_{-i},\,v)}{P(h_{-i},\,v)}v_j \\
&= -\,\frac{P(h_i = 1, h_{-i},\,v)}{P(h_i = 1, h_{-i},\,v) + P(h_i = 0, h_{-i},\,v)}v_j \\
&= -\,\frac{\frac{1}{Z}e^{-\frac{E_\theta(h_i=1,h_{-i},v)}{T}}}{\frac{1}{Z}e^{-\frac{E_\theta(h_i=1,h_{-i},v)}{T}} + \frac{1}{Z}e^{-\frac{E_\theta(h_i=0,h_{-i},v)}{T}}}v_j \\
&= -\,\frac{1}{1 + e^{-\frac{E_\theta(h_i=0,h_{-i},v)}{T}+\frac{E_\theta(h_i=1,h_{-i},v)}{T}}}v_j \\
&= -\,\frac{1}{1 + e^{\frac{[-\beta(v,h_{-i})+0\cdot\alpha_k(v)]}{T}+\frac{[\beta(v,h_{-i})+1\cdot\alpha_k(v)]}{T}}}v_j \\
&= -\,\frac{1}{1 + e^{-\frac{\alpha_i(v)}{T}}}v_j
\end{aligned} \tag{9}$$

Then, it is obtained that

$$\begin{aligned}
\frac{\ln\ \partial P(v)}{\partial w_{i,j}} &= \sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(\boldsymbol{v},\,h)}{\partial w_{i,j}} - \sum_v P(v)\sum_h P(h|v)\frac{\partial E_\theta(v,\,h)}{\partial \theta} \\
&= \sum_h P(h|\boldsymbol{v})h_i v_j + \sum_v P(v)P(h_i = 1|v)v_j \\
&= \frac{1}{1 + e^{-\frac{\alpha_i(\boldsymbol{v})}{T}}}\boldsymbol{v}_j - \sum_v P(v)\frac{1}{1 + e^{-\frac{\alpha_i(v)}{T}}}v_j \\
&\approx \frac{1}{1 + e^{-\frac{\alpha_i(\boldsymbol{v})}{T}}}\boldsymbol{v}_j - \frac{1}{1 + e^{-\frac{\alpha_i\left(v^k\right)}{T}}}v_j^k
\end{aligned} \tag{10}$$

where v_j^k is the k–th Gibbs sampling of v_j. Similarly, for the case that $\theta = a_i$ and $\theta = b_i$, we obtain that

$$\begin{aligned}
\frac{\ln\ \partial P(v)}{\partial a_i} &= -\sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(\boldsymbol{v},\,h)}{\partial a_i} + \sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(\boldsymbol{v},\,h)}{\partial a_i} \\
&\approx \boldsymbol{v}_i - v_i^k
\end{aligned} \tag{11}$$

and

$$\begin{aligned}
\frac{\ln\ \partial P(v)}{\partial b_i} &= -\sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(\boldsymbol{v},\,h)}{\partial b_i} + \sum_h P(h|\boldsymbol{v})\frac{\partial E_\theta(\boldsymbol{v},\,h)}{\partial b_i} \\
&= P(h_i = 1|\boldsymbol{v}) - P(h_i = 1|v^k) \\
&\approx \frac{1}{1 + e^{-\frac{\alpha_i(\boldsymbol{v})}{T}}} - \frac{1}{1 + e^{-\frac{\alpha_i\left(v^k\right)}{T}}}
\end{aligned} \tag{12}$$

It can be seen that T actually controls the sharpness of the logistic function from equations (10) to (12), and the learning algorithm is given by:

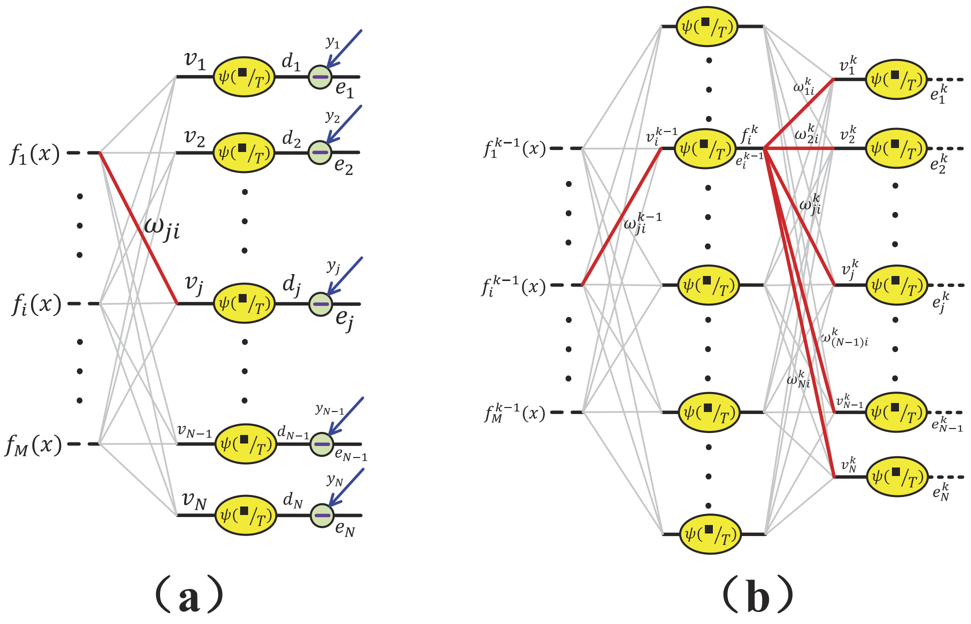

Figure 2. Illustration of the back propagation on TRBMs.

$$
\begin{aligned}
w_{i,j}(n+1) &= w_{i,j}(n) + \eta \cdot \frac{\ln\ \partial P(v)}{\partial w_{i,j}} \\
a_j(n+1) &= a_j^n(n) + \eta \cdot \frac{\ln\ \partial P(v)}{\partial a_j^n} \\
b_j(n+1) &= b_j^n(n) + \eta \cdot \frac{\ln\ \partial P(v)}{\partial b_j^n}
\end{aligned}
\tag{13}
$$

Thus, this theorem holds. □

Theorem 1 indicates that the effect of the temperature parameter can be effectively reflected on the sharpness of the logistic sigmoid function. This benefits the implementation of contrastive divergence in pre-training a TRBM as one only needs to adjust T as seen in equations (10)–(12).

Back propagation for fine-tuning a TRBM. In employing the contrastive divergence algorithm for pre-training a TRBM, we have shown that the sharpness of the logistic sigmoid function reflects the temperature effection. In the fine-tuning stage, the back propagation will be applied. In this section, we further show how the temperature parameter affect the back propagation progress for fine-tuning a TRBM. It is shown that the error propagated from the output layer will be multiplied by $\frac{1}{T}$ in every layer.

Let the logistic sigmoid function, which is also called the activation function, of the TRBM be

$$\psi(x/T) = \frac{1}{1 - e^{-x/T}} \tag{14}$$

Note that $0 < \psi(x/T) < 1$ and the derivative of a sigmoid function is

$$
\begin{aligned}
\frac{d\psi(x/T)}{dx} &= \frac{1}{T} \cdot \psi(x/T) \cdot (1 - \psi(x/T)) \\
&= \frac{1}{T}\widetilde{\psi}(x/T)
\end{aligned}
\tag{15}
$$

where $\widetilde{\psi}(x/T) = \psi(x/T) \cdot (1 - \psi(x/T))$ and we have $0 < \widetilde{\psi}(x/T) < 1$.

Theorem 2. When applying back propagation for fine-tuning a TRBM, the error signal propagated from the output layer of TRBM will be multiplied by $\frac{1}{T}$ at every layer.

Proof. From Fig. 2(a), when considering the gradient on the output layer, the cost function $R = \sum_j e_j^2$, where $e_j = y_j - d_j$, d_j is the output of the network and y_j is the given labels. We have

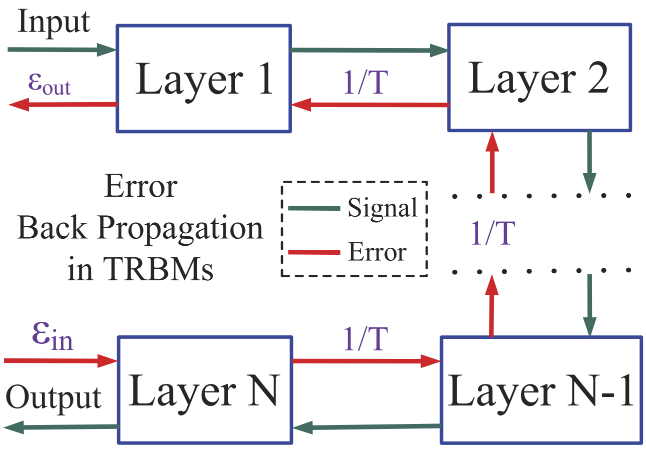

Figure 3. Illustrate of how temperature affects the back propagation progress. The error propagated from the output layer will be multiplied by $1/T$ in every layer. For a relative higher temperature since $T > 1$, the amplitude of the gradient will be reduced by $1/T$ times. For a relative higher temperature $T < 1$, the amplitude of the gradient will be strengthened by $1/T$ times.

$$
\begin{aligned}
\frac{dR}{dw_{ji}^{(out)}} &= \frac{R}{e_j}\frac{e_j}{d_j}\frac{d_j}{v_j}\frac{v_j}{w_{ji}^{(out)}}\\
&= e_j \cdot (-1) \cdot \frac{1}{T} \cdot \widetilde{\psi}(v_j) \cdot f_i(x)\\
&= -\frac{1}{T} \cdot e_j \cdot \widetilde{\psi}(v_j) \cdot f_i(x)
\end{aligned}
\tag{16}
$$

As shown in Fig. 2(b), suppose that the error signal on the layer κ which is transformed back from its next layer is fixed as e_j^{κ}, the cost function on layer κ can be regarded as $R^{\kappa} = \sum_j \left[e_j^{\kappa}\right]^2$. Then, the gradient of the cost function with respect to the weight $w_{ji}^{(\kappa-1)}$ on layer $\kappa - 1$ by fixing the error signal e_j^{κ} on layer κ is given by

$$
\begin{aligned}
\frac{dR^{\kappa}}{dw_{ji}^{(\kappa-1)}} &= \sum_j \frac{R^{\kappa}}{e_j^{\kappa}} \cdot \frac{e_j^{\kappa}}{v_j^{\kappa}} \cdot \frac{v\kappa_j}{f_i^{\kappa}} \cdot \frac{f_i^{\kappa}}{v_i^{\kappa-1}} \cdot \frac{v_i^{\kappa-1}}{w_{ji}^{(k-1)}}\\
&= \sum_j e_j^{\kappa} \cdot \frac{1}{T} \cdot \widetilde{\psi}\left(v_j^{\kappa}\right) w_{ji}^{\kappa} \cdot \frac{1}{T} \cdot \widetilde{\psi}\left(v_j^{\kappa-1}\right) \cdot f_i^{\kappa-1}(x)\\
&= \frac{1}{T} \cdot e_i^{\kappa-1} \cdot \widetilde{\psi}\left(v_j^{\kappa-1}\right) \cdot f_i^{\kappa-1}(x)
\end{aligned}
\tag{17}
$$

where

$$
e_i^{\kappa-1} = \frac{1}{T} \cdot \sum_j e_j^{\kappa} \cdot \widetilde{\psi}\left(v_j^{\kappa}\right) w_{ji}^{\kappa}
\tag{18}
$$

is the error signal for the neuron i on layer $\kappa - 1$, and it is the summarization of all the error signals transferred from layer κ. Comparing with the standard back propagation progress in RBMs, the error in doing back propagation in TRBM will be multiplied by $\frac{1}{T}$, as summarized in Fig. 3. □

Simulations

A 784-500-500-2000-10 network is built after layer-by-layer pre-training of TRBMs and overall fine-tuning under the back propagation algorithm. the training data is a MNIST set which contains 60000 handwritten digits. As there is no parameter T in a RBM and for the convenience of comparisons we introduce another parameter T_0 set such that $T/T_0 = 1$ corresponds the standard RBM. So the values of T/T_0 reflect how we set the temperature in TRBM, i.e., a higher T/T_0 implies a relative higher temperature compared with a RBM. We achieve 8 groups of weight parameters by training the network at different temperatures respectively, including T/T_0 of 0.1, 0.2, 0.5 0.8, 1.0, 1.2, 1.5 and 2.0. For each testing, 10000 groups of sampling data of the firing neuron number in layer 2 are computed, in which the input for each sampling is randomly chosen from the 10000 digits in the MNIST testing set. Notably, the firing neuron is also a result of probabilistic sampling. For example, if the output of a neuron in layer 2 is 0.3, it has a chance of thirty percent to fire. At last, we illustrate the neuronal activity distribution via drawing the histogram of 10000 groups of the firing neuron number in layer 2. As the temperature gradually rises, the activity distribution curve moves to the right which indicates more firing neurons; while, the reverse result is observed when the temperature decreases.

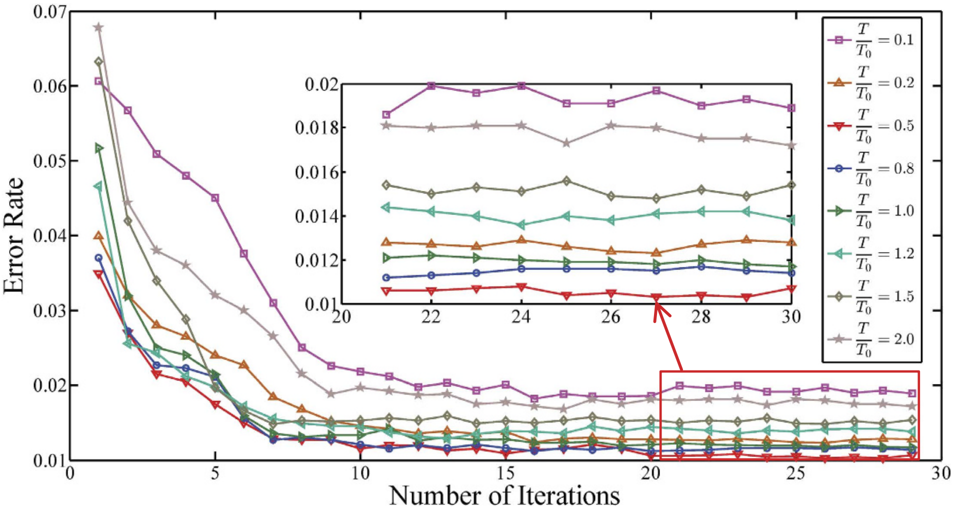

Figure 4. The performance of the TRBMs with respect to different temperatures. We choose 8 groups of weight parameters by training the network at different temperatures respectively, including T/T_0 of 0.1, 0.2, 0.5 0.8, 1.0, 1.2, 1.5 and 2.0.

Figure 4 shows the performance of the TRBMs with respect to different temperature values. We allow T/T_0 changing from 0.1 to 2. Note that when $T/T_0 = 1$, the model is reduced to the standard RBM. The temperature increases as $T/T_0 > 1$, and decreases as $T/T_0 < 1$. It is observed that the performance of the TRBM improves as the temperature decreases significantly when T/T_0 is higher than 1. And it can be further improved when T/T_0 becomes lower and achieves its best performance around $T/T_0 = 0.5$. After that, the performance deteriorates as $T/T0$ decreases. The best error rate is around 1%.

To see how temperature affects the extracted features on layer 2, we plot these features in Fig. 5. Note that each neuron in layer 2 has a 784-dimensional weight vector as it connects 784 neurons in layer 1. We could reshape the 784-dimensional weight vector to a 28 × 28 matrix which is called a reconstruction map. As the reconstruction map can be reconstructed for each neuron in layer 2, all the maps we reconstructed are considered as extracted features on this layer. Figure 5 shows 15 × 15 randomly chosen reconstruction maps, where each block element corresponds to a reconstruction map of a neuron in layer 2. The reconstruction maps are obtained at four different temperatures: (a) $T/T_0 = 0.1$; (b) $T/T_0 = 0.5$; (c) $T/T_0 = 1.0$; (d) $T/T_0 = 2.0$. Four typical features are reconstructed, including blank, stroke, digit and snowflake. The blank feature indicates small weights and snowflake feature is resulted from large weights. As temperature arises, i.e., T/T_0 increases, blank features become less while more snowflake features appear, which leads to intense responses of the post-neurons. Note that when $T/T_0 = 1.0$ in Fig. 5(c), the TRBM is reduced to the standard RBM, where three features including stroke, digit and snowflake can be observed. But the TRBM captures the best stroke features at a relatively lower temperature $T/T_0 = 0.5$ in Fig. 2(b), where the blank, the digit and the snowflake are almost invisible. However, as the temperature falls continuously, sparse activities[64] of the neurons lead to better selectivity, but more inactive connections, i.e. blank features.

In addition, we show how the temperature affect the extracted features on the third layer in Fig. 6. Each neuron in layer 3 has a 500-dimensional weight vector connected to the neurons in layer 2, and each neuron in layer 2 has a weight reconstruction map. Consequently, we can reconstruct a weight map of each neuron in layer 3, by computing a weighted sum of all 500 reconstruction maps of the neurons in layer 2. Similarly, we obtain the extracted features on layer 3 at four different temperatures: (a) $T/T_0 = 0.1$; (b) $T/T_0 = 0.5$; (c) $T/T_0 = 1.0$; (d) $T/T_0 = 2.0$. Since the digit features are usually expected to appear at higher layers, here we focus on the number of digit features in the weight reconstruction map at different temperatures. As the temperature rises, the digit features significantly decrease. Considering the digit features in Figs 4 and 5 in the meantime, we conclude that a lower temperature accelerates the travelling speed of digit features from the input layer to the output layer. For example, when $T/T_0 = 0.1$, the digit features begin to appear in the weight reconstruction map of the neurons in layer 2; while, this phenomenon is still not obvious even in the weight reconstruction map of the neurons in layer 2 when $T/T_0 = 2$.

We also estimate the probability distribution of the number of firing neurons (or called "neuron activity distribution") under different temperatures. Firstly, it is worthy to make clarifications on parameter T as the operation of TRBM is not the same as a physical heat exchange process. Since T can be considered as a mathematical parameter, it can be fixed as a constant when using the proposed modified back propagation (BP) algorithm to investigate its effects. In this case, the network runs by repeatedly choosing a unit and setting its state based on the training algorithm. According to a Boltzmann distribution, after running for sufficiently long time with a fixed T, the probability of a state of the network will depend only upon the final energy level of that state, and not on the initial state from which the process is started. This relationship indicates that the state distribution has converged to an equilibrium state. For example, as shown in Fig. 7 in the paper, if we train the network at a lower temperature T_1, the probability distribution of the number of firing neurons (neuron activity distribution) will converge

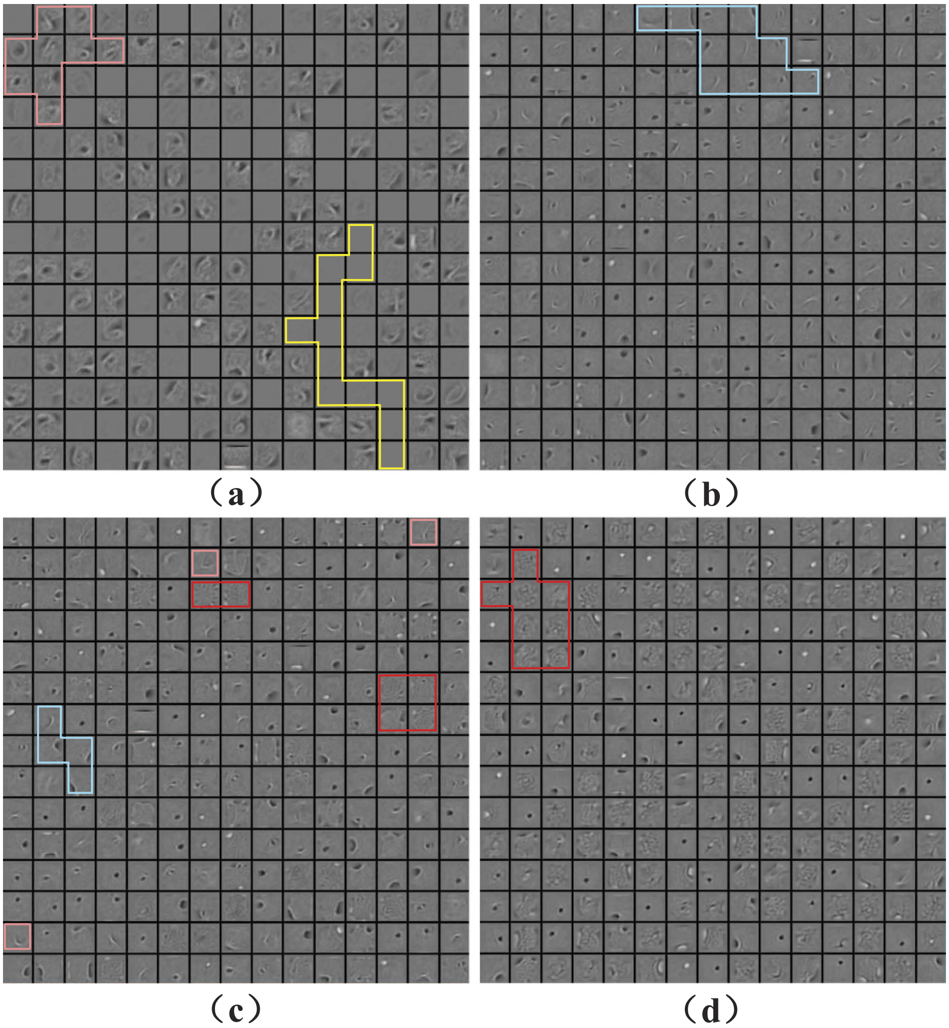

Figure 5. Temperature effection on the extracted features on layer 1 for (**a**) $T/T_0 = 0.1$; (**b**) $T/T_0 = 0.5$; (**c**) $T/T_0 = 1.0$; (**d**) $T/T_0 = 2.0$.

to an equilibrium state, i.e. the left-most curve; while if we increase the temperature to T_3 when training, the state will converge to a new equilibrium state, i.e. the right-most curve. This implies that, the network state will reach a particular equilibrium distribution for a particular given parameter T. In other words, there exists a one-one mapping between T and the neuron activity distribution.

How T affects the firing neuron activity distribution can be tested by repeating experiments with a different fixed T for each training process with the results given in Fig. 7. It is observed that parameter T affects the firing neurons activity distribution similar to that of a temperature parameter in Boltzmann distribution illustrated in Fig. 1. Particularly, with lower temperature, the neuron activity decreases but with higher entropy and selectivity; while higher temperature leads to intense neuron activity. So the previous mentioned "equilibrium distribution" can be reasonably considered as a similar real-life thermal equilibrium. This is also why we could treat the parameter T as a temperature, and name our newly proposed RBM as Temperature based Restricted Boltzmann Machines (TRBM).

The kurtosis[65] can be applied to characterize the selectivity of the TRBM, and it is the degree of peakedness of a distribution, defined as a normalized form of the fourth central moment of a distribution. Generally speaking, a higher degree of peakedness corresponds to a better selectivity. For a random variable X with mean and variance being μ and σ^2, respectively, the kurtosis is defined by

$$Kurt(X) = \frac{E((X-\mu)^4))}{\sigma^4} - 3 \tag{19}$$

The "minus 3" at the end of this formula is often explained as a correction to make the kurtosis of the normal distribution equal to zero. It is observed that a higher temperature also leads to a more flat distribution, which has a lower kurtosis, i.e. poor selectivity. In contrast, better selectivity of low power often results in smaller error rate for pattern recognition. However, if the temperature is too low, the neuronal activity will be so sparse that the

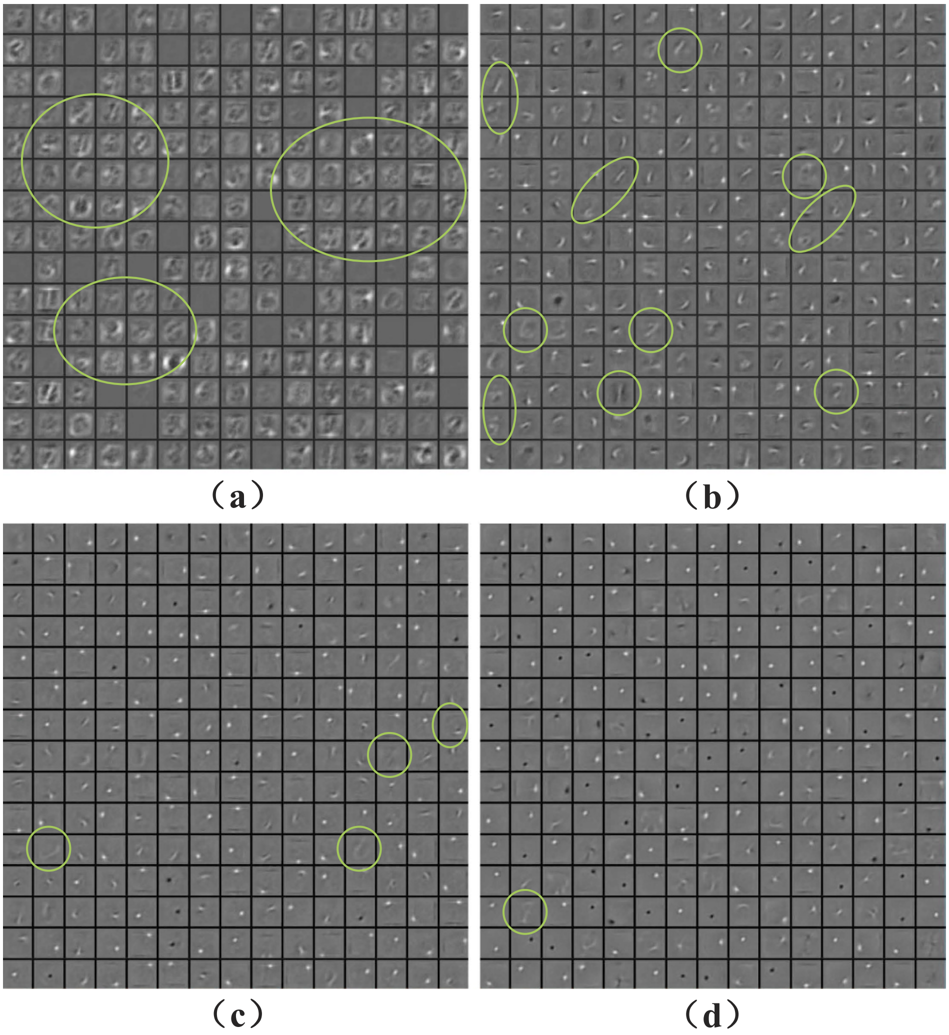

Figure 6. Temperature effection on the extracted features on layer 3 for (a) $T/T_0 = 0.1$; **(b)** $T/T_0 = 0.5$; **(c)** $T/T_0 = 1.0$; **(d)** $T/T_0 = 2.0$.

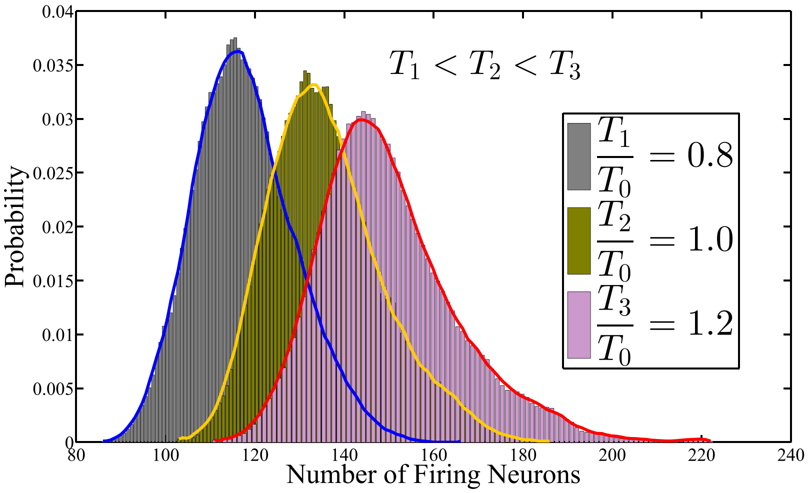

Figure 7. Estimated probability distribution of the number of firing neurons under different temperatures. The results are obtained by repeating the experiments with a different fixed T for each training process.

recognition results will become worse. In conclusion, for a relatively lower temperature, the distributions moves to the left side with a higher kurtosis, which implies that a lower temperature leads to a lower particle activity but a higher entropy. This is consistent with the observations on the neuron activity distribution in real-life physical systems in Fig. 1.

Conclusions and Discussions

In this work, we propose a temperature based restricted Boltzmann machines which reveals that temperature is an essential parameter for controlling the selectivity of the firing neurons in the hidden layers. The two theorems we have established reveal the simplicity and applicability when implementing the proposed TRBM, because the effect of temperature can be efficiently adjusted by setting the sharpness of the logistic function, and the error propagated from the output layer only needs to be multiplied by $\frac{1}{T}$ in every layer during the back propagation process. Clearly, our work brings more benefits to RBMs by bringing temperature into the model, such as more flexible choices, more completed and accurate modelling and results.

On the other hand, the work in this paper allows us to understand how temperature affects the performance of the TRBMs. It also stimulates our curiosity and opens our mind to think deeply about whether temperature affects the cognitive performance of real-life neural systems. An interesting study on cognitive performance of human beings in different seasons was conducted[66], where tested subjects are required to respond to a visual stimulus with a key press as quickly as possible. Researchers found that the reaction of the subjects is faster in winter than in Summer, as they are more focused in cold weather. This interesting phenomenon consists with our observations in this paper, namely, relatively lower temperature improves the performance of a neural systems. Because the neural activity becomes thinner (higher kurtosis) in a relatively lower temperature environment, and thinner neural activity distribution makes the system have more feature selectivity ability. However, we know that more and more neurons will be inactive as the temperature decreases continually. Therefore, the performance improvement only exists in a narrow interval. The biological evolution has adjusted the neural systems to work well in a proper temperature range, which may have small fluctuations with respect to temperature. This provides a more comprehensive understanding on the artificial neural systems from a physical point of view, and may be essential for investigating the biological and artificial intelligence.

References

1. Xu, J., Li, H. & Zhou, S. An overview of deep generative models. *IETE Tech. Rev.* **32,** 131–139 (2015).
2. Langkvist, M. & Loutfi, A. Learning feature representations with a cost-relevant sparse autoencoder. *Int. J. Neural Syst.* **25,** 1450034 (2015).
3. Zhang, G. *et al.* An optimization spiking neural P system for approximately solving combinatorial optimization problems. *Int. J. Neural Syst.* **24,** 1440006 (2014).
4. Schmidhuber, J. Deep learning in neural networks: an overview. *Neural Networks* **117,** 85–117 (2015).
5. Schneider, R. & Card, H. C. Instabilities and oscillation in the deterministic Boltzmann machine. *Int. J. Neural Syst.* **10,** 321–330 (2000).
6. Chen, L. H. *et al.* Voice conversion using deep neural networks with layer-wise generative training. *IEEE/ACM Trans. Audio, Speech, and Language Process.* **22,** 1859–1872 (2014).
7. Fischer, A. & Igel, C. Training restricted Boltzmann machines: an introduction. *Pattern Recogn.* **25,** 25–39 (2014).
8. Smolensky, P. Information processing in dynamical systems: foundations of harmony theory. *Parallel Distributed Processing* 1, 194–281 (MIT-Press 1986).
9. Hinton, G. E. & Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks, *Science* **313,** 504–507 (2006).
10. Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural Comput.* **14,** 1771–1800 (2002).
11. Hinton, G. E. & Salakhutdinov, R. R. Discovering binary codes for documents by learning deep generative models. *Top. Cogn. Sci.* **3,** 74–91 (2011).
12. Fischer, A. & Igel, C. An introduction to restricted Boltzmann machines. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications* 7441, 14–36, Buenos Aires, Argentina. Springer Berlin Heidelberg. (doi: 10.1007978-3-642-33275-3_2) (2012).
13. Larochelle, H. & Bengio, Y. Classification using discriminative restricted Boltzmann machines. *Proc. 25th International Conference on Machine Learning* 536–543, Helsinki, Finland. ACM New York, NY, USA. (doi: 10.1145/1390156.1390224) (2008).
14. Zhang, C. X. Learning ensemble classifiers via restricted Boltzmann machines. *Pattern Recogn. Lett.* **36,** 161–170 (2014).
15. Hayat, M., Bennamoun, M. & An, S. Deep reconstruction models for image set classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **37,** 713–727 (2015).
16. Li, Q. *et al.* Credit risk classification using discriminative restricted Boltzmann machines. *Proc. 17th International Conference on Computational Science and Engineering* 1697–1700, Chengdu, China. (doi: 10.1109/CSE.2014.312) (2014).
17. An, X. *et al.* A deep learning method for classification of EEG data based on motor imagery. *Proc. 10th International Conference on Intelligent Computing: Intelligent Computing in Bioinformatics* 203–210, Taiyuan, China. Springer International Publishing. (doi: 10.1007/978-3-319-09330-7_25) (2014).
18. Chen, F. *et al.* Spectral classification using restricted Boltzmann machine. *Publ. Astron. Soc. Aust.* **31,** e001 (2014).
19. Coates, A., Ng, A. Y. & Lee, H. An analysis of single-layer networks in unsupervised feature learning. *Proc. 14th International Conference on Artificial Intelligence and Statistics* 215–223, Fort Lauderdale, FL, USA. (2011).
20. Suk, H. I. *et al.* Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage* **101,** 569–582 (2014).
21. Xie, J. Learning features from high speed train vibration signals with deep belief networks. *International Joint Conference on Neural Networks* 2205–2210, Beijing, China. (doi: 10.1109/IJCNN.2014.6889729) (2014).
22. Nie, L., Kumar, A. & Zhan, S. Periocular recognition using unsupervised convolutional RBM feature learning. *IEEE 22nd International Conference on Pattern Recognition* 399–404, Stockholm, Sweden. (doi: 10.1109/ICPR.2014.77) (2014).
23. Huang, Z. *et al.* Speech emotion recognitionwith unsupervised feature learning. *Front. Inform. Technol. Electron. Eng.* **16,** 358–366 (2015).
24. Huynh, T., He, Y. & Rger, S. Learning higher-level features with convolutional restricted Boltzmann machines for sentiment analysis. *Proc. 37th European Conference on IR Research* 447–452, Vienna, Austria. (doi: 10.1007/978-3-319-16354-3_49) (2015).
25. Campbell, A., Ciesielksi, V. & Qin, A. K. Feature discovery by deep learning for aesthetic analysis of evolved abstract images. *Proc. 4th International Conference on Evolutionary and Biologically Inspired Music, Sound, Art and Design* 27–38, Copenhagen, Denmark. (doi: 10.1007/978-3-319-16498-4_3) (2015).

26. Ji, N. *et al.* Discriminative restricted Boltzmann machine for invariant pattern recognition with linear transformations. *Pattern Recogn. Lett*. **45,** 172–180 (2014).
27. Chen, G. & Srihari, S. N. A noisy-or discriminative restricted Boltzmann machine for recognizing handwriting style development. *IEEE 14th International Conference on Frontiers in Handwriting Recognition* 714–719, Heraklion, Greece. (doi: 10.1109/ICFHR.2014.125) (2014).
28. Li, G. *et al.* Behind the magical numbers: hierarchical chunking and the human working memory capacity. *Int. J. Neural Syst.* **24,** 1350019 (2013).
29. Jia, X. *et al.* A novel semi-supervised deep learning framework for affective state recognition on EEG signals. *IEEE 14th International Conference on Bioinformatics and Bioengineering* 30–37, Boca Raton, FL, USA. (doi: 10.1109/BIBE.2014.26) (2014).
30. Hinton, G. E. & Salakhutdinov, R. R. Replicated softmax: an undirected topic model. *Advances in Neural Information Processing Systems* 1607–1614 (2009).
31. Zieba, M., Tomczak, J. M. & Gonczarek, A. RBM-SMOTE: restricted Boltzmann machines for synthetic minority oversampling technique. *Proc. 7th Asian Conference: Intelligent Information and Database Systems* 377–386, Bali, Indonesia. (doi: 10.1007/978-3-319-15702-3_37) (2015).
32. Kuremoto, T. *et al.* Time series forecasting using a deep belief network with restricted Boltzmann machines. *Neurocomputing* **137,** 47–56 (2014).
33. Hjelm, R. D. *et al.* Restricted Boltzmann machines for neuroimaging: An application in identifying intrinsic networks. *NeuroImage* **96,** 245–260 (2014).
34. Sakai, Y. & Yamanishi, K. Data fusion using restricted Boltzmann machines. *IEEE International Conference on Data Mining* **2014**, 953–958 (2014).
35. Jian, S. L. *et al.* SEU-tolerant restricted Boltzmann machine learning on DSP-based fault detection. *IEEE 12th International Conference on Signal Processing* 1503–1506, Hangzhou, China. (doi: 10.1109/ICOSP.2014.7015250) (2014).
36. Sheri, A. M. *et al.* Contrastive divergence for memristor-based restricted Boltzmann machine. *Eng. Appl. Artif. Intel*. **37,** 336–342 (2015).
37. Goh, H. *et al.* Unsupervised and supervised visual codes with restricted Boltzmann machines. *Proc. 12th European Conference on Computer Vision* 298–311, Florence, Italy. (doi: 10.1007/978-3-642-33715-4_22) (2012).
38. Plis, S. M. *et al.* Deep learning for neuroimaging: a validation study. *Front. Neurosci*. **8,** 229 (2014).
39. Pedroni, B. U. *et al.* Neuromorphic adaptations of restricted Boltzmann machines and deep belief networks, *IEEE International Joint Conference on Neural Networks* 1–6, Dallas, TX, USA. (doi: 10.1109/IJCNN.2013.6707067) (2013).
40. Landau, L. D. & Lifshitz, E. M. Statistical physics. *Course of Theoretical Physics* **5,** 468 (1980).
41. Mendes, G. A. *et al.* Nonlinear Kramers equation associated with nonextensive statistical mechanics. *Phys. Rev. E* **91,** 052106 (2015).
42. e Silva, L. B. *et al.* Statistical mechanics of self-gravitating systems: mixing as a criterion for indistinguishability. *Phys. Rev. D* **90,** 123004 (2014).
43. Gadjiev, B. & Progulova, T. Origin of generalized entropies and generalized statistical mechanics for superstatistical multifractal systems. *International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering* **1641,** 595–602 (2015).
44. Boozer, A. D. Thermodynamic time asymmetry and the Boltzmann equation. *Am. J. Phys*. **83,** 223 (2015).
45. Tang, H. Y., Wang, J. H. & Ma, Y. L. A mew approach for the statistical thermodynamic theory of the nonextensive systems confined in different finite traps. *J. Phys. Soc. Jpn*. **83,** 064004 (2014).
46. Shim, J. W. & Gatignol, R. Robust thermal boundary condition using Maxwell-Boltzmann statistics and its application. *AIP Conference Proceedings-American Institute of Physics* **1333**, 980 (2011).
47. Gordon, B. L. Maxwell-Boltzmann statistics and the metaphysics of modality. *Synthese* **133,** 393–417 (2002).
48. Niven, R. K. Exact Maxwell-Boltzmann, Bose-Einstein and Fermi-Dirac statistics. *Phys. Lett. A* **342,** 286–293 (2005).
49. Lin, H. *et al.* Curvelet domain denoising based on kurtosis characteristics. *J. Geophys. Eng.* **12,** 419–426 (2015).
50. Bekenstein, J. D. Black holes and entropy. *Phys. Rev. D* **7,** 2333 (1973).
51. Rrnyi, A. On measures of entropy and information. *Fourth Berkeley Symposium on Mathematical Statistics and Probability* **1,** 547–561 (1961).
52. Li, J., Li, J. & Yan, S. Multi-instance learning using information entropy theory for image retrieval. *17th IEEE International Conference on Computational Science and Engineering* 1727–1733, Chengdu, China. (doi: 10.1109/CSE.2014.317) (2014).
53. Reed, L. J. & Berkson, J. The application of the logistic function to experimental data. *The Journal of Physical Chemistry* **33,** 760–779 (1929).
54. Chen, Z., Cao, F. & Hu, J. Approximation by network operators with logistic activation functions. *Appl. Math. Comput.* **256,** 565–571 (2015).
55. Hastings, W. K. Monte Carlo sampling methods using Markov Chains and their applications. *Biometrika* **57,** 97–109 (1970).
56. Green, P. J. Reversible jump Markov Chain Monte Carlo computation and bayesian model determination. *Biometrika* **82,** 711–732 (1995).
57. Derin, H. & Kelly, P. Discrete-index Markov-type random processes. *Proc. IEEE* **77,** 1485–1510 (1989).
58. Keiding, N. & Gill, R. D. Random truncation models and Markov processes. *Ann. Stat*. **18,** 582–602 (1990).
59. Bengio, Y. & Delalleau, O. Justifying and generalizing contrastive divergence. *Neural Comput.* **21,** 1601–1621 (2009).
60. Neftci, E. *et al.* Event-driven contrastive divergence for spiking neuromorphic systems. *Front. Neurosci*. **7,** 272 (2014).
61. Sanger, T. D. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks* **2,** 459–473 (1989).
62. Kolmogorov, V. & Zabih, R. What energy functions can be minimized via graph cuts? *IEEE Trans. Pattern Anal. Mach. Intell.* **26,** 147–159 (2004).
63. Elfwing, S., Uchibe, E. & Doya, K. Expected energy-based restricted Boltzmann machine for classification. *Neural Networks* **64,** 29–38 (2015).
64. Boureau, Y. & Cun, Y. L. Sparse feature learning for deep belief networks. *Advances in Neural Information Processing Systems* 1185–1192 (2008).
65. Kenney, J. F. & Keeping, E. S. *Mathematics of Statistics*. (Princeton, NJ: Van Nostrand 1951).
66. Brennen, T. *et al.* Arctic cognition: a study of cognitive performance in summer and winter at 69°N. *Appl. Cognitive Psych*. **13,** 561–580 (1999).

Acknowledgements

This work was supported in part by the Independent research plan of Tsinghua University (Grant No. 20141080934 and No. 20151080467), and the National Natural Science Foundation of China (Grant No. 61475080), and the Science and Technology Plan of Beijing, China (Grant No. Z151100000915071).

Author Contributions

G.L., L.D., Y.X. and C.W. proposed the model. L.D., W.W., J.P. and L.S. designed the experiments and simulations. G.L. and C.W. proved the Theorems. All authors write the manuscript.

Additional Information

Competing financial interests: The authors declare no competing financial interests.

How to cite this article: Li, G. *et al.* Temperature based Restricted Boltzmann Machines. *Sci. Rep.* **6**, 19133; doi: 10.1038/srep19133 (2016).