Exponential Family Restricted Boltzmann Machines and Annealed Importance Sampling

Conference Paper · July 2018

DOI: 10.1109/IJCNN.2018.8489413

2 authors, including:

Yifeng Li
National Research Council Canada
63 PUBLICATIONS **428** CITATIONS

SEE PROFILE

Exponential Family Restricted Boltzmann Machines and Annealed Importance Sampling

Yifeng Li
Digital Technologies Research Centre
National Research Council Canada
Ottawa, Ontario, Canada
Email: yifeng.li@nrc-cnrc.gc.ca

Xiaodan Zhu
Department of Electrical and Computer Engineering
Queen's University
Kingston, Ontario, Canada
Email: xiaodan.zhu@queensu.ca

***Abstract*—In this paper, we investigate restricted Boltzmann machines (RBMs) from the exponential family perspective, enabling the visible units to follow any suitable distributions from the exponential family. We derive a unified view to compute the free energy function for exponential family RBMs (exp-RBMs). Based on that, annealed important sampling (AIS) is generalized to the entire exponential family, allowing for estimating the log-partition function and log-likelihood. Our experiments on a document processing task demonstrate that the generalized free energy functions and AIS estimation perform well in helping capture useful knowledge from the data; the estimated log-partition functions are stable. The appropriate instances of exp-RBMs can generate novel and meaningful samples and can be applied to classification tasks.**

***Keywords*-deep learning, generative model, exponential family, restricted Boltzmann machine, annealed importance sampling**

I. Introduction

In the landscape of probabilistic distributions, discriminative models, such as convolutional or recurrent networks, are more interested in the domain boundaries rather than the actual distributions. This naturally makes them suitable for tasks such as classification and regression. A generative model, however, is concerned with the joint distribution $p(\boldsymbol{x}, \boldsymbol{h})$, where $\boldsymbol{x}$ generally represents observed measurements including class labels (if any), and $\boldsymbol{h}$ denotes hidden variables (if any). Even though generative models can be adapted for classification or regression, they could also be used in a larger versatility of applications. For example, when class labels are unavailable, generative models are unsupervisedly learnable on a large amount of data. Second, generative models can generate novel samples, which is beneficial to creative tasks. Furthermore, generative models allow us to infer the values of unobserved variables, which can be applied on clustering, dimensionality reduction, multivariate regression, pattern completion, inference, etc. The learned joint distributions potentially allow us to address a variety of AI challenges [1].

Typical generative neural networks include logistic belief net (LBN) [2], Helmholtz machine (HM) [3], restricted Boltzmann machine (RBM) [4], deep belief net (DBN) [5], and deep Boltzmann machines (DBM) [6]. Inferring the hidden states given observed data (i.e. $p(\boldsymbol{h}|\boldsymbol{x})$) remains the core interest and challenge in generative models. Since the logistic belief net only has generative connections, inference is approximated by time-consuming Markov chain Monte Carlo sampling. In addition to generative connections, the Helmholtz machine allows stochastic recognition connections to approximate $p(\boldsymbol{h}|\boldsymbol{x})$. Both sets of parameters are updated using a wake-sleep algorithm, a variant of stochastic gradient descent. RBM is an energy-based model as the joint distribution is presented using an energy function. The inference in RBM is exact, as the homogeneous connections are ignored. Stacking an RBM and an HM together makes a DBN which can be learned using greedy layer-wise pretraining and fine-tuning using a wake-sleep algorithm. The pretraining alleviates the stigma of parameter learning in deep models and reignites the deep learning research [7]. DBM can be viewed as a stack of RBMs. However, the inference in DBM is intractable but can be approximated by the mean-field variational algorithm. A unified theory of modelling, learning and inference in other distributions is still the major bottleneck preventing generative models from being the first consideration.

One common feature of the aforementioned models is that a neuron (say z_i, either from $\boldsymbol{x}$ or $\boldsymbol{h}$) is limited to the binary type which is activated with probability $\sigma(b_i + \boldsymbol{w}_i^{\mathrm{T}}\Pi(z_i))$, where $\sigma(\cdot)$ is the sigmoid function, b_i is the bias on z_i, $\Pi(z_i)$ is the parent nodes of z_i, and $\boldsymbol{w}_i$ contains the weights of connections from $\Pi(z_i)$ to z_i. Essentially, RBM is the foundation of the above deep generative models. Although the binary RBM has been adapted to accommodate Gaussian signals [8] and word counts [9], these *ad hoc* models are yet to be explained in the theory of exponential family harmonium or RBM (exp-RBM) that is merely very briefly discussed in [10] where a set of key questions remain unanswered. These questions include: (1) How are the conditional distributions derived and represented in both natural parameter space and standard parameter space? (2) Is it possible to compute the free energy functions of exp-RBMs? And (3), can we estimate the partition functions and the log-likelihood of the exp-RBMs?

To address these questions, this paper contributes from the following perspectives: (1) we re-examine and revise the exponential family RBM, derive the conditionals in the natural and standard parametric spaces, derive the generic stochastic parameter learning rules, and provide instances for selected distributions; (2) we derive the general equation to efficiently compute the free energy functions for exp-RBMs; (3) we

generalize the annealed importance sampling (AIS) method to estimate the log-partition functions of exp-RBMs and use them to estimate the corresponding log-likelihood.

II. Background and Notations

A. Binary Restricted Boltzmann Machine

Boltzmann machine (BM) falls into the energy-based probabilistic model class where a model can be formulated as

$$p(\boldsymbol{x},\boldsymbol{h}) = \frac{1}{Z}e^{-E(\boldsymbol{x},\boldsymbol{h})}, \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{h}$ represent respective vectors of visible and hidden variables, $E(\boldsymbol{x},\boldsymbol{h})$ is the energy function, and Z the partition function. As a special case of BM [11], RBM ignores the homogeneous interactions, making the conditionals decomposable [4]. In RBM, the energy function is defined as

$$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}, \tag{2}$$

where $\boldsymbol{x} \in \{0,1\}^M$ is a vector of binary visible variables, $\boldsymbol{h} \in \{0,1\}^K$ is a vector of hidden variables, $\boldsymbol{a}$ and $\boldsymbol{b}$ are biases, and $\boldsymbol{W}$ governs the heterogeneous interactions.

The decomposability of conditionals comes from the decomposability of its energy function that can be rewritten to

$$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{T}\boldsymbol{x} + \sum_{k=1}^{K}\Big(-b_k h_k - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h_k\Big). \tag{3}$$

Hence, the conditional $p(\boldsymbol{h}|\boldsymbol{x})$ can be factorized into:

$$\begin{aligned} p(\boldsymbol{h}|\boldsymbol{x}) &= \frac{p(\boldsymbol{x},\boldsymbol{h})}{p(\boldsymbol{x})} = \frac{e^{\boldsymbol{a}^{T}\boldsymbol{x}+\sum_{k=1}^{K}(b_k h_k + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h_k)}}{\sum_{\boldsymbol{h}'} e^{\boldsymbol{a}^{T}\boldsymbol{x}+\sum_{k=1}^{K}(b_k h'_k + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h'_k)}} \\ &= \frac{\prod_{k=1}^{K} e^{b_k h_k + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h_k}}{\prod_{k=1}^{K}\sum_{h'_k} e^{b_k h'_k + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h'_k}} = \prod_{k=1}^{K} p(h_k|\boldsymbol{x}), \end{aligned} \tag{4}$$

where $\sum_{x_1,\cdots,x_K} f(x_1)\cdots f(x_K) = \prod_{k=1}^{K}\sum_{x_k} f(x_k)$ is used. Since $h_k \in \{0,1\}$, the conditional is computed as

$$p(h_k = 1|\boldsymbol{x}) = \frac{e^{b_k h_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}h_k}}{\sum_{h'_k} e^{b_k h'_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}h'_k}} = \sigma(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}). \tag{5}$$

Similar to Eq. (4), $p(\boldsymbol{x}|\boldsymbol{h})$ is also decomposable:

$$p(\boldsymbol{x}|\boldsymbol{h}) = \prod_{m=1}^{M} p(x_m|\boldsymbol{h}), \tag{6}$$

where $x_m \in \{0,1\}$. We have

$$p(x_m = 1|\boldsymbol{h}) = \sigma(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}). \tag{7}$$

This leads to a Gibbs method to sample from $p(\boldsymbol{x},\boldsymbol{h})$:

$$\boldsymbol{h}|\boldsymbol{x} \sim \sigma(\boldsymbol{b} + \boldsymbol{W}^{\mathrm{T}}\boldsymbol{x});\ \boldsymbol{x}|\boldsymbol{h} \sim \sigma(\boldsymbol{a} + \boldsymbol{W}\boldsymbol{h}). \tag{8}$$

The model parameters is estimated by maximizing the likelihood $p(\boldsymbol{x})$ that is obtained by marginalizing out $\boldsymbol{h}$:

$$p(\boldsymbol{x}) = \sum\!\!\!\!\!\!\int_{\boldsymbol{h}'} p(\boldsymbol{x},\boldsymbol{h}') = \frac{1}{Z}\sum\!\!\!\!\!\!\int_{\boldsymbol{h}'} e^{-E(\boldsymbol{x},\boldsymbol{h}')} = \frac{1}{Z}e^{-F(\boldsymbol{x})}, \tag{9}$$

where $F(\boldsymbol{x}) = -\log \sum\!\!\!\!\!\!\int_{\boldsymbol{h}'} e^{-E(\boldsymbol{x},\boldsymbol{h}')}$ is the free energy function, and $Z = \sum\!\!\!\!\!\!\int_{\boldsymbol{x}'} e^{-F(\boldsymbol{x}')} = \sum\!\!\!\!\!\!\int_{\boldsymbol{x}'}\sum\!\!\!\!\!\!\int_{\boldsymbol{h}'} e^{-E(\boldsymbol{x}',\boldsymbol{h}')}$. If $E(\boldsymbol{x},\boldsymbol{h})$ can be expressed by an additive combination

$$E(\boldsymbol{x},\boldsymbol{h}) = -\zeta(\boldsymbol{x}) + \sum_{k=1}^{K}\gamma_k(\boldsymbol{x},h_k), \tag{10}$$

then $F(\boldsymbol{x})$ (and the likelihood) can be computed tractably:

$$F(\boldsymbol{x}) = -\zeta(\boldsymbol{x}) - \sum_{k=1}^{K}\log\sum\!\!\!\!\!\!\int_{h_k} e^{-\gamma_k(\boldsymbol{x},h_k)}. \tag{11}$$

Thus, from Eq. (3), the free energy function of binary RBM can be computed analytically and efficiently:

$$F(\boldsymbol{x}) = -\boldsymbol{a}^{T}\boldsymbol{x} - \sum_{k=1}^{K}\log\sum_{h_k} e^{b_k h_k + \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}_{:,k}h_k}. \tag{12}$$

The model parameters of the binary RBM include $\boldsymbol{\theta} = \{\boldsymbol{a},\boldsymbol{b},\boldsymbol{W}\}$, which can be estimated by the stochastic gradient descent algorithm. To do so, we minimize the averaged negated log-likelihood of a set of samples $\boldsymbol{X} = [\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N]$:

$$l(\boldsymbol{X};\boldsymbol{\theta}) = \frac{1}{N}\sum_{n=1}^{N}\big(F(\boldsymbol{x}_n) + \log Z\big). \tag{13}$$

Its first-order derivative is

$$\frac{\partial l(\boldsymbol{X};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = \frac{1}{N}\sum_{n=1}^{N}\Big(\frac{\partial F(\boldsymbol{x}_n)}{\partial\boldsymbol{\theta}} - \mathrm{E}_{p(\boldsymbol{x})}\Big[\frac{\partial F(\boldsymbol{x})}{\partial\boldsymbol{\theta}}\Big]\Big), \tag{14}$$

where one can find that the key is to derive $\frac{\partial F(\boldsymbol{x})}{\partial\boldsymbol{\theta}}$:

$$\frac{\partial F(\boldsymbol{x})}{\partial\boldsymbol{\theta}} = \frac{\partial(-\log\sum\!\!\!\!\!\!\int_{\boldsymbol{h}'} e^{-E(\boldsymbol{x},\boldsymbol{h}')})}{\partial\boldsymbol{\theta}} = \mathrm{E}_{p(\boldsymbol{h}|\boldsymbol{x})}\Big[\frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\boldsymbol{\theta}}\Big]. \tag{15}$$

Thus, the first-order derivative (gradient) of the averaged negated log-likelihood with respective to $\boldsymbol{\theta}$ has the form

$$\Delta_{\boldsymbol{\theta}} = \frac{1}{N}\sum_{n=1}^{N}\Big(\mathrm{E}_{p(\boldsymbol{h}|\boldsymbol{x}_n)}\Big[\frac{\partial E(\boldsymbol{x}_n,\boldsymbol{h})}{\partial\boldsymbol{\theta}}\Big] - \mathrm{E}_{p(\boldsymbol{x},\boldsymbol{h})}\Big[\frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\boldsymbol{\theta}}\Big]\Big), \tag{16}$$

where the first term is a data-dependent term, and the second a data-independent term. Usually the maximal likelihood estimation is solved by gradient descent algorithms, where the parameter is iteratively updated using the rule $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \epsilon\Delta_{\boldsymbol{\theta}}$ where $\epsilon > 0$ is the learning rate. From Eq. (16), one knows that $\frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\theta}$ is of key importance to compute the gradient. In virtue of the simplicity of binary RBM's energy function, one can elegantly obtain the first-order derivatives of $E(\boldsymbol{x},\boldsymbol{h})$ with respect to $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{W}$, respectively:

$$\frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\boldsymbol{a}} = -\boldsymbol{x};\ \frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\boldsymbol{b}} = -\boldsymbol{h};\ \frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial\boldsymbol{W}} = -\boldsymbol{x}\boldsymbol{h}^{\mathrm{T}}. \tag{17}$$

The posterior in Eq. (4) is used to compute the data-dependent term in Eq. (16). Gibbs sampling [Eq. (8)] is used in the (persistent) k-step contrastive divergence (CD-k) [12], [13] algorithm to draw fantasies from the joint distribution for the data-independent term.

During and after the learning, the log-likelihood

$$\log p(\boldsymbol{x}) = -F(\boldsymbol{x}) - \log Z \tag{18}$$

is a measure of learning quality. The free energy can be computed using Eq. (12) for binary RBM. Its computation for other distributions is generally viewed to be unaffordable. However, we shall show that this view is incorrect. Furthermore, the partition function for binary RBM is estimated using the AIS method [14]. But, how to estimate partition functions of RBMs with other distributions remains unknown before our work.

B. Exponential Family

The exponential family includes univariate or multivariate distributions with the following natural parametric form:

$$p(\boldsymbol{x}) = h(\boldsymbol{x})e^{\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{s}(\boldsymbol{x})-A(\boldsymbol{\theta})}, \tag{19}$$

where $\boldsymbol{x}$ is either a univariate or multivariate random variable, $\boldsymbol{\theta}$ is a vector (or a scalar) of natural parameters, $\boldsymbol{s}(\boldsymbol{x})$ is a vector (or a scalar) of sufficient statistics, $A(\boldsymbol{\theta})$ is the log-partition function, and $h(\boldsymbol{x})$ is the base measure. The exponential family has several interesting properties. One can easily see from below that $A(\boldsymbol{\theta})$ normalizes the distribution:

$$A(\boldsymbol{\theta}) = \log\Big[\sum_{\boldsymbol{x}}\!\!\!\!\!\int h(\boldsymbol{x})e^{\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{s}(\boldsymbol{x})}\Big]. \tag{20}$$

Reformulating Eq. (19) to $p(\boldsymbol{x}) = \frac{1}{Z}e^{\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{s}(\boldsymbol{x})+\log h(\boldsymbol{x})}$, one finds $\log Z = A(\boldsymbol{\theta})$, therefore the name of $A(\boldsymbol{\theta})$. In latter sections, one will see that $A(\boldsymbol{\theta})$ is extremely useful in the estimation of the partition functions of exp-RBMs.

The exponential class includes Bernoulli, Gaussian, Poisson distributions, etc. It also accommodates some members with certain parameters fixed, such as negative binomial distribution with fixed number of successful Bernoulli trials, and multinomial distribution with fixed total number of counts. For the understanding of this article, we selectively provide the natural and standard forms of a range of widely used univariate distributions in Table I, which is very helpful and frequently referred in latter contents.

There are constraints in some of the natural forms which must also be fulfilled in the parameter estimation of exp-RBMs. For instance, in the natural form of Gaussian distribution, θ_2 must be negative, if not fixed *a posteriori*. In negative binomial distribution, since $\theta = \log(1-p)$, θ must remain negative. In multinoulli and multinomial distributions, we do not require θ to be negative in the natural forms. Instead, to fulfill $\sum_{m=1}^{M} p_m = 1$ in the standard forms, we apply the softmax function such that $p_m = \frac{e^{\theta_m}}{\sum_{m'=1}^{M} e^{\theta_{m'}}}$.

III. Method

The binary RBM only allows binary visible and hidden variables. It has been extended in [10] to allow distributions from the exponential family. Since the original description of exp-RBMs is too concise and difficult for general readers to follow, here we revisit and revise the general procedure of constructing an exp-RBM, derive the conditionals for Gibbs sampling, and build several instances that will be highly useful in practice. Most importantly, we explicitly generalize the free energy function and the AIS method to the exponential family in order to estimate the log-partition function and log-likelihood.

A. Generic Procedure of Building an Exp-RBM

In the design of an exp-RBM, three key components are required: (1) the assumed independent prior distributions of visible variable $\boldsymbol{x}$ and hidden variable $\boldsymbol{h}$, (2) the energy function $E(\boldsymbol{x},\boldsymbol{h})$, and (3) the conditional distributions $p(\boldsymbol{x}|\boldsymbol{h})$ and $p(\boldsymbol{h}|\boldsymbol{x})$. These components are elaborated below.

First, we should define independent prior distributions for both $\boldsymbol{x}$ and $\boldsymbol{h}$ (while their interactions are temporally assumed to be null) in the natural forms of exponential class:

$$p(\boldsymbol{x}) = \prod_{m=1}^{M} e^{\boldsymbol{a}_m^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)-A_m(\boldsymbol{a}_m)} \tag{21}$$

$$p(\boldsymbol{h}) = \prod_{k=1}^{K} e^{\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{t}_k+\log g_k(h_k)-B_k(\boldsymbol{b}_k)}. \tag{22}$$

From the concept of exponential class, $\boldsymbol{a}_m$ and $\boldsymbol{b}_k$ are vectors of natural parameters for x_m and h_k respectively, $\boldsymbol{s}_m$ and $\boldsymbol{t}_k$ are corresponding vectors of sufficient statistics, $A_m(\boldsymbol{a}_m)$ and $B_k(\boldsymbol{b}_k)$ are corresponding log-partition functions, and $f_m(x_m)$ and $g_k(h_k)$ are corresponding base measures. Depending on specific distributions, these vectors of natural parameters and sufficient statistics may be reduced to scalars. For example, if $p(\boldsymbol{x})$ is Gaussian, then $\boldsymbol{a}_m = [a_m^{(1)}, a_m^{(2)}]^{\mathrm{T}} = [\mu_m\beta_m, -\frac{\beta_m}{2}]^{\mathrm{T}}$ and $\boldsymbol{s}_m = [s_m^{(1)}, s_m^{(2)}]^{\mathrm{T}} = [x_m, x_m^2]^{\mathrm{T}}$ which are 2-dimensional vectors. If $p(\boldsymbol{h})$ is Bernoulli, then $b_k = \log\frac{p_k}{1-p_k}$ and $t_k = h_k$ are scalars.

Second, the energy function $E(\boldsymbol{x},\boldsymbol{h})$ of the exp-RBM can be defined via (i) combining x_m-related terms in Eq. (21) and h_k-related terms in Eq. (22) (including base measures), and (ii) enabling interactions between $\boldsymbol{s}_m$ and $\boldsymbol{t}_k$:

$$\begin{aligned} E(\boldsymbol{x},\boldsymbol{h}) = &-\sum_{m=1}^{M}\left(\boldsymbol{a}_m^{\mathrm{T}}\boldsymbol{s}_m + \log f_m(x_m)\right) - \sum_{k=1}^{K}\left(\boldsymbol{b}_k^{\mathrm{T}}\boldsymbol{t}_k + \log g_k(h_k)\right) \\ &- \sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{r=1}^{R}\sum_{u=1}^{U} s_{m,r}w_{m,k,r,u}t_{k,u}, \end{aligned} \tag{23}$$

where R and U are the numbers of natural parameters for $p(x_m)$ and $p(h_k)$. The interaction strengths are represented by a tensor $\boldsymbol{W} \in \mathbb{R}^{M\times K\times R\times U}$ where a slice $\boldsymbol{W}_{:,:,r,u}$ is a matrix that is also denoted by $\boldsymbol{W}^{(r,u)}$. For Gaussian-Bernoulli RBM (exp-RBM with Gaussian visible variables and Bernoulli hidden variables), the weight tensor is of size $M \times K \times 2$, including $\boldsymbol{W}^{(1)}$ for interactions between $\boldsymbol{x}$ and $\boldsymbol{h}$, and $\boldsymbol{W}^{(2)}$ for $\boldsymbol{x}^{*2}$ and $\boldsymbol{h}$. To reduce model complexity, we may consider to drop off some interactions between the sufficient statistics. For instance, in Gaussian-Bernoulli RBM, we may decide to disregard $\boldsymbol{W}^{(2)}$, reducing the weights in a tensor to a matrix of size $M \times K$. In fact, we only and always consider interactions between $\boldsymbol{x}$ and $\boldsymbol{h}$ in practice, reducing the interaction term to $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$.

Third, from the energy function defined in Eq. (23), we can derive the conditionals as below. Using the definition of exponential family in Eq. (19), $p(\boldsymbol{x}|\boldsymbol{h})$ can be derived as

TABLE I: Selected members of the exponential family.

Distribution	Standard Form	Natural Form	$\boldsymbol{\theta}(\boldsymbol{\eta})$	$\boldsymbol{\eta}(\boldsymbol{\theta})$	$\boldsymbol{s}(x)$	$h(x)$	$A(\boldsymbol{\eta})$	$A(\boldsymbol{\theta})$	Sufficient Statistics
Gaussian	$p(x\|\mu,\lambda^{-1}) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}(x-\mu)^2}, \lambda > 0$	$p(x\|\boldsymbol{\theta}) = e^{\theta_1 s_1 + \theta_2 s_2 - A(\boldsymbol{\theta})}, \theta_2 < 0$	$\begin{bmatrix} \mu\lambda \\ -\frac{\lambda}{2} \end{bmatrix}$	$\begin{bmatrix} -\frac{\theta_1}{2\theta_2} \\ -2\theta_2 \end{bmatrix}$	$\begin{bmatrix} x \\ x^2 \end{bmatrix}$	1	$\frac{1}{2}\log\frac{2\pi}{\lambda} + \frac{\mu^2\lambda}{2}$	$\frac{1}{2}\log\frac{\pi}{-\theta_2} - \frac{\theta_1^2}{4\theta_2}$	$\mathrm{E}[x] = \mu = -\frac{\theta_1}{2\theta_2}$ $\mathrm{Var}[x] = \frac{1}{\lambda} = -\frac{1}{2\theta_2}$ $\mathrm{E}[x^2] = \mu^2 + \frac{1}{\lambda} = \frac{\theta_1^2}{4\theta_2^2} + \frac{1}{-2\theta_2}$
Gaussian (fix precision λ)	$p(x\|\mu,\lambda^{-1}) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}(x-\mu)^2}, \lambda > 0$	$p(x\|\theta) = (\frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}x^2}) e^{\theta s - A(\boldsymbol{\theta})}$	μ	θ	λx	$\frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}x^2}$	$\frac{\mu^2\lambda}{2}$	$\frac{\theta^2\lambda}{2}$	$\mathrm{E}[x] = \mu = \theta$ $\mathrm{Var}[x] = \frac{1}{\lambda}$ $\mathrm{E}[x^2] = \mu^2 + \frac{1}{\lambda} = \theta^2 + \frac{1}{\lambda}$
Gaussian (fix precision λ)	$p(x\|\mu,\lambda^{-1}) = \frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}(x-\mu)^2}, \lambda > 0$	$p(x\|\theta) = (\frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}x^2}) e^{\theta s - A(\boldsymbol{\theta})}$	$\mu\lambda$	$\frac{\theta}{\lambda}$	x	$\frac{\sqrt{\lambda}}{\sqrt{2\pi}} e^{-\frac{\lambda}{2}x^2}$	$\frac{\mu^2\lambda}{2}$	$\frac{\theta^2}{2\lambda}$	$\mathrm{E}[x] = \mu = \frac{\theta}{\lambda}$ $\mathrm{Var}[x] = \frac{1}{\lambda}$ $\mathrm{E}[x^2] = \mu^2 + \frac{1}{\lambda} = \frac{\theta^2}{\lambda^2} + \frac{1}{\lambda}$
Poisson	$p(x\|\lambda) = \frac{e^{-\lambda}}{x!}\lambda^x, \lambda > 0, x \geq 0$	$p(x\|\theta) = \frac{1}{x!} e^{\theta s - A(\theta)}$	$\log\lambda$	e^θ	x	$\frac{1}{x!}$	λ	e^θ	$\mathrm{E}[x] = \lambda = e^\theta$ $\mathrm{Var}[x] = \lambda = e^\theta$
Bernoulli	$p(x\|p) = p^x(1-p)^{1-x}, p \in (0,1), x \in \{0,1\}$	$p(x\|\theta) = e^{\theta s - A(\theta)}$	$\log\frac{p}{1-p}$	$\frac{e^\theta}{1+e^\theta} = \sigma(\theta)$	x	1	$-\log(1-p)$	$-\log(1-\sigma(\theta)) = \log(1+e^\theta)$	$\mathrm{E}[x] = p = \sigma(\theta)$ $\mathrm{Var}[x] = p(1-p) = \sigma(\theta)(1-\sigma(\theta))$
Binomial (fix number of trials n)	$p(x\|n,p) = \binom{n}{x} p^x(1-p)^{n-x}, p \in (0,1), x \geq 0$	$p(x\|\theta) = \frac{n!}{x!(n-x)!} e^{\theta s - A(\theta)}$	$\log\frac{p}{1-p}$	$\frac{e^\theta}{1+e^\theta} = \sigma(\theta)$	x	$\frac{n!}{x!(n-x)!}$	$-n\log(1-p)$	$-n\log(1-\sigma(\theta))$	$\mathrm{E}[x] = np = n\sigma(\theta)$ $\mathrm{Var}[x] = np(1-p) = n\sigma(\theta)(1-\sigma(\theta))$
Negative Binomial (fix number of successes k, success rate p)	$p(x\|k,p) = \binom{x+k-1}{k-1} p^k(1-p)^x, p \in (0,1), x \geq 0$	$p(x\|\theta) = \frac{(x+k-1)!}{(k-1)!x!} e^{\theta s - A(\theta)}, \theta < 0$	$\log(1-p)$	$1 - e^\theta$	x	$\frac{(x+k-1)!}{(k-1)!x!}$	$-k\log p$	$-k\log(1-e^\theta)$	$\mathrm{E}[x] = k\frac{1-p}{p} = k\frac{e^\theta}{1-e^\theta}$ $\mathrm{Var}[x] = k\frac{1-p}{p^2} = k\frac{e^\theta}{(1-e^\theta)^2}$
Multinoulli	$p(x_1,\cdots,x_M\|p_1,\cdots,p_M) = p_1^{x_1}\cdots p_M^{x_M}$, $p_m \geq 0$, $\sum_{m=1}^M p_m = 1$, $x_m \in \{0,1\}$, $\sum_{m=1}^M x_m = 1$	$p(x\|\boldsymbol{\theta}) = e^{\theta_1 s_m + \cdots + \theta_M s_M - \log C}$, where $C = \sum_{m=1}^M e^{\theta_m}$	$\begin{bmatrix} \log p_1 + \log C \\ \vdots \\ \log p_M + \log C \end{bmatrix}$	$\begin{bmatrix} \frac{e^{\theta_1}}{C} \\ \vdots \\ \frac{e^{\theta_M}}{C} \end{bmatrix}$	$\begin{bmatrix} x_1 \\ \vdots \\ x_M \end{bmatrix}$	1	$\log C$	$\log C$	$\mathrm{E}[x_m = 1] = p_m = \frac{e^{\theta_m}}{C}$ $\mathrm{Var}[x_m = 1] = p_m(1-p_m) = \frac{e^{\theta_m}}{C-\theta_m}$
Multinomial (fix number of trials n)	$p(x_1,\cdots,x_M\|n,p_1,\cdots,p_M) = \frac{n!}{\prod_{m=1}^M x_m!} p_1^{x_1}\cdots p_M^{x_M}, p_m \geq 0, \sum_{m=1}^M p_m = 1, x_m \in \{0,1,\cdots,n\}, \sum_{m=1}^M x_m = n$	$p(x\|\theta) = \frac{n!}{\prod_{m=1}^M x_m!} e^{\theta_1 s_m + \cdots + \theta_M s_M}$ where $C = \sum_{m=1}^M e^{\theta_m}$	$\begin{bmatrix} \log p_1 + \log C \\ n\log C \\ \log p_M + \log C \end{bmatrix}$	$\begin{bmatrix} \frac{e^{\theta_1}}{C} \\ \vdots \\ \frac{e^{\theta_M}}{C} \end{bmatrix}$	$\begin{bmatrix} x_1 \\ \vdots \\ x_M \end{bmatrix}$	$\frac{n!}{\prod_{m=1}^M x_m!}$	$n\log C$	$n\log C$	$\mathrm{E}[x_m] = np_m = n\frac{e^{\theta_m}}{C}$ $\mathrm{Var}[x] = np_m(1-p_m) = n\frac{e^\theta}{C-e^\theta}$

$$
\begin{aligned}
p(\boldsymbol{x}|\boldsymbol{h}) &= \frac{p(\boldsymbol{x},\boldsymbol{h})}{p(\boldsymbol{h})} = \frac{e^{-E(\boldsymbol{x},\boldsymbol{h})}}{\int_{\boldsymbol{x}} e^{-E(\boldsymbol{x},\boldsymbol{h})}} \\
&= \frac{e^{\sum_{m=1}^{M}\left(\boldsymbol{a}_m^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)\right)+\sum_{m=1}^{M}\sum_{k=1}^{K}(\boldsymbol{s}_m)^{\mathrm{T}}\boldsymbol{W}_{m,k,:,:}\boldsymbol{t}_k}}{\int_{\boldsymbol{x}} e^{\sum_{m=1}^{M}\left(\boldsymbol{a}_m^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)\right)+\sum_{m=1}^{M}\sum_{k=1}^{K}(\boldsymbol{s}_m)^{\mathrm{T}}\boldsymbol{W}_{m,k,:,:}\boldsymbol{t}_k}} \\
&= \frac{e^{\sum_{m=1}^{M}\left(\hat{\boldsymbol{a}}_m^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)-A_m(\hat{\boldsymbol{a}}_m)\right)}}{\int_{\boldsymbol{x}} e^{\sum_{m=1}^{M}\left(\hat{\boldsymbol{a}}_m^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)-A_m(\hat{\boldsymbol{a}}_m)\right)}} \\
&= e^{\sum_{m=1}^{M}\left((\hat{\boldsymbol{a}}_m)^{\mathrm{T}}\boldsymbol{s}_m+\log f_m(x_m)-A_m(\hat{\boldsymbol{a}}_m)\right)} \\
&= \prod_{m=1}^{M} e^{\hat{\boldsymbol{a}}_m^{\mathrm{T}}\boldsymbol{s}_m(x_m)+\log f_m(x_m)-A_m(\hat{\boldsymbol{a}}_m)} \\
&= \prod_{m=1}^{M} p(x_m|\boldsymbol{h},\boldsymbol{\eta}(\hat{\boldsymbol{a}}_m)), \qquad (24)
\end{aligned}
$$

where $\hat{\boldsymbol{a}}_m = \boldsymbol{a}_m + \sum_{k=1}^{K}\boldsymbol{W}_{m,k,:,:}\boldsymbol{t}_k$, and function $\boldsymbol{\eta}(\hat{\boldsymbol{a}}_m)$ maps the natural parameters in $\hat{\boldsymbol{a}}_m$ to the standard forms. As shown in Table I, if $p(x_m|\boldsymbol{h})$ is Gaussian, we then have $\boldsymbol{\eta}(\hat{\boldsymbol{a}}_m) = [\hat{\mu}_m, \hat{\beta}_m]^{\mathrm{T}} = [-\frac{\hat{a}_m^{(1)}}{2\hat{a}_m^{(2)}}, -2\hat{a}_m^{(2)}]^{\mathrm{T}}$. Similarly, we have the conditional over the hidden variables:

$$
p(\boldsymbol{h}|\boldsymbol{x}) = \prod_{k=1}^{K} e^{\hat{\boldsymbol{b}}_k^{\mathrm{T}}\boldsymbol{t}_k+\log g_k(h_k)-B_k(\hat{\boldsymbol{b}}_k)} = \prod_{k=1}^{K} p(h_k|\boldsymbol{x},\boldsymbol{\eta}(\hat{\boldsymbol{b}}_k)), \quad (25)
$$

where $\hat{\boldsymbol{b}}_k = \boldsymbol{b}_k + \sum_{m=1}^{M}(\boldsymbol{W}_{m,k,:,:})^{\mathrm{T}}\boldsymbol{s}_m$ and $\boldsymbol{\eta}(\hat{\boldsymbol{b}}_k)$ maps the natural parameters in $\hat{\boldsymbol{b}}_k$ to the standard ones. For example, if $p(h_k|\boldsymbol{x})$ is Bernoulli, then we have $\eta(\hat{b}_k) = \hat{p}_k = \sigma(\hat{b}_k)$. The key point here is that the conditional distributions are decomposable and following the same forms as the prior distributions, but with posterior parameters reflecting the influence of their heterogeneous counterparts.

Stochastic gradient descent algorithm can be used to estimate the parameters in exp-DBM. The model parameters of exp-RBM can also be represented by $\boldsymbol{\theta} = \{\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{W}\}$ where $\boldsymbol{a} = \{\boldsymbol{a}^{(1)}, \cdots, \boldsymbol{a}^{(R)}\}$ and $\boldsymbol{b} = \{\boldsymbol{b}^{(1)}, \cdots, \boldsymbol{b}^{(U)}\}$ may respectively include several bias vectors, and $\boldsymbol{W} = \{\boldsymbol{W}^{(1,1)}, \cdots, \boldsymbol{W}^{(r,u)}, \cdots, \boldsymbol{W}^{(R,U)}\}$ may be a collection of several interaction matrices (but we, throughout the rest of this paper, only consider the interactions between $\boldsymbol{x}$ and $\boldsymbol{h}$). To estimate these parameters, the general form of gradient in Eq. (16) is still applicable. We needs to derive the first-order derivatives of $E(\boldsymbol{x},\boldsymbol{h})$ [Eq. (23)] w.r.t. the parameters:

$$
\frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial a_m^{(r)}} = -s_m^{(r)}; \ \frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial b_k^{(u)}} = -t_k^{(u)}; \ \frac{\partial E(\boldsymbol{x},\boldsymbol{h})}{\partial w_{m,k}^{(r,u)}} = -s_m^{(r)}t_k^{(u)}. \quad (26)
$$

The generic concepts of exp-RBM presented above differ from the original work [10] in three points: (1) base measures are added in the energy function, so that we can conveniently derive the conditionals; (2) the new conditionals also contain base measures and are formulated in both natural and standard spaces to instantiate epx-RBMs and their Gibbs samplings (see below); and (3) learning rules are derived from joint distribution rather than likelihood, consistent with binary RBM.

The traditional binary RBM is a special case of exp-RBM, that is Bernoulli-Bernoulli RBM. We can design instances of exp-RBMs with visible and hidden variables following any appropriate distributions from the exponential family. Here, we selectively enumerate several instances of exp-RBMs in Table II that are quite useful in applications. Most of them adopt Bernoulli hidden variables, because in the potential exp-RBM-based pretraining of a supervised deep model, the Bernoulli distribution of hidden variable in (stochastic) exp-RBM is related to the sigmoid activation function of the (deterministic) deep discriminative model. Below are a few important remarks regarding the instantiation of exp-RBMs (Table II). (1) The natural forms (rather than standard forms) of parameters should be used in the bias terms of the exp-RBM models. However, in the presentation of conditional distributions, we need to use the standard forms, because they are involved in Gibbs sampling. (2) The biases and weights in some models have to be constrained. For instance, in Negative-Binomial-Bernoulli RBM, a_m is defined as the logarithm of failure probability, hence must be always negative. Furthermore, its conditional $p(x_m|\boldsymbol{h})$ has a failure rate $e^{\hat{a}_m} = e^{a_m+\boldsymbol{W}_{m,:}\boldsymbol{h}}$. Thus, the posterior parameter $\hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ must be negatively constrained as well. This leads to a projection stochastic gradient descent algorithm, where we must keep $a_m < 0$ and $\hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h} < 0$ ($\forall \boldsymbol{h} \in \{0,1\}^K$). Thus we should use the projection rules $a_m = \min\{a_m, -\epsilon\}$ and $w_{m,k} = \min\{w_{m,k}, 0\}$, where ϵ is a very tiny positive number (say 10^{-10}). (3) Some interaction terms may be ignored to improve efficiency or compatibility when pretraining other generative or discriminative models. In Gaussian-Bernoulli RBM, the second interaction term $(\boldsymbol{x}^{*2})^{\mathrm{T}}\boldsymbol{W}^{(2)}\boldsymbol{h}$ may be dropped off. Alternatively, we can pre-specify and fix the precision in Gaussian distribution. Following the generic definition of exp-RBM, the three Gaussian-Bernoulli RBMs given in Table II is consistent with the *ad-hoc* forms of Gaussian-Bernoulli RBMs discussed in [15], [16], [17]. (4) Although exp-RBM uses stochastic gradient descent as in the traditional binary RBM for parameter learning, the exact learning rules could be different among exp-RBMs, depending on the number of natural parameters and exact form of sufficient statistics. (5) Modern data often contain counts, such as word counts in document modelling and read count per DNA fragment in next-generation genome sequencing. Poisson, negative binomial and multinomial distributed visible units meet this need. (6) Categorical types are common in survey data and in classification, where multinoulli distribution is applicable. (7) The replicated softmax RBM [9] is in fact a variant of multinomial-Bernoulli RBM in the framework of exp-RBM, repeating a multinoulli-Bernoulli RBM (with only one visible multinoulli unit) for multiple times. To address the issue of variable document length, for a sample with count depth n, the energy function in the original replicated softmax RBM is defined as $E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - n\boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where multiplying $\boldsymbol{b}^{\mathrm{T}}\boldsymbol{h}$ by n is equivalent to normalizing $\boldsymbol{x}$ to have unit l_1-norm ($c = 1$), while we decide to normalize all documents to have the same count $c \geq 1$ before running multinomial-Bernoulli RBM.

B. Generalizing the Free Energy and AIS Method

Computing the partition functions of exp-RBMs are generally viewed to be impossible [10]. Here we shall show that the log-partition functions can be estimated by using our

TABLE II: Instances of exp-RBM.

Model	Energy Function	Conditional	Gradient
Bernoulli-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_M}{1-p_M}]^{\mathrm{T}}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{BE}(x_m\vert\sigma(\hat{a}_m)),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(h_k\vert\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Gaussian-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -(\boldsymbol{a}^{(1)})^{\mathrm{T}}\boldsymbol{x} - (\boldsymbol{a}^{(2)})^{\mathrm{T}}\boldsymbol{x}^{\boldsymbol{*2}} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a}^{(1)} = [\mu_1\lambda_1, \cdots, \mu_M\lambda_M]^{\mathrm{T}}$, $\boldsymbol{a}^{(2)} = [\frac{-\lambda_1}{2}, \cdots, \frac{-\lambda_M}{2}]^{\mathrm{T}}, \boldsymbol{a}^{(2)} < \boldsymbol{0}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{GS}(x_m\vert -\frac{\hat{a}_m^{(1)}}{2a_m^{(2)}}, (-2a_m^{(2)})^{-1}),\ \hat{a}_m^{(1)} = a_m^{(1)} + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(h_k\vert\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}^{(1)}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{a}^{(2)}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n^{*2}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s^{*2}\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Gaussian-Bernoulli fix precision $\boldsymbol{\lambda}$, model 1	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}(\boldsymbol{\lambda}*\boldsymbol{x}) - \sum_{m=1}^{M}(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2) - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - (\boldsymbol{\lambda}*\boldsymbol{x})^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\mu_1, \cdots, \mu_M]^{\mathrm{T}}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{GS}(x_m\vert\hat{a}_m, \lambda_m),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(h_k\vert\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}(\boldsymbol{\lambda}*\boldsymbol{x})$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{\lambda}*\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{\lambda}*\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-(\boldsymbol{\lambda}*\boldsymbol{x}_n)\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-(\boldsymbol{\lambda}*\boldsymbol{x}_s)\boldsymbol{h}_s^{\mathrm{T}}\right)$
Gaussian-Bernoulli fix precision $\boldsymbol{\lambda}$, model 2	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2) - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\mu_1\lambda_1, \cdots, \mu_M\lambda_M]^{\mathrm{T}}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{GS}(x_m\vert\frac{\hat{a}_m}{\lambda_m}, \lambda_m),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(h_k\vert\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Poisson-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\log\lambda_1, \cdots, \log\lambda_M]^{\mathrm{T}}$, $\boldsymbol{b}_k = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]^{\mathrm{T}}$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{PO}(x_m\vert e^{\hat{a}_m}),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(h_k\vert\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Poisson-Binomial	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \sum_{k=1}^{K}\log\frac{n!}{h_k!(n-h_k!)} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\log\lambda_1, \cdots, \log\lambda_M]^{\mathrm{T}}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]^{\mathrm{T}}$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{PO}(x_m\vert e^{\hat{a}_m}),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BN}(n_k, \sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Negative-Binomial-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}\log\frac{(x_m+s_m-1)!}{(s_m-1)!x_m!} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\log p_1, \cdots, \log p_M]^{\mathrm{T}}, \boldsymbol{a} < \boldsymbol{0}$ $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]^{\mathrm{T}}$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \prod_{m=1}^{M}\mathcal{NB}(x_m\vert s_m, 1-e^{\hat{a}_m}),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h} < 0$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$
Multinoulli-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -\sum_{m=1}^{M}\boldsymbol{a}^{(m)\mathrm{T}}\boldsymbol{x}^{(m)} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \sum_{m=1}^{M}(\boldsymbol{x}^{(m)})^{\mathrm{T}}\boldsymbol{W}^{(m)}\boldsymbol{h}$, where $\boldsymbol{a}^{(m)} = [\log p_1^{(m)} + \log C^{(m)}, \cdots, \log p_{C_m}^{(m)} + \log C^{(m)}]^{\mathrm{T}}$, $C_m = \text{size}(\boldsymbol{x}^{(m)}), C^{(m)} = \sum_{c'=1}^{C_m}\exp(a_{c'}^{(m)})$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]^{\mathrm{T}}$, $\boldsymbol{W}^{(m)} \in \mathbb{R}^{C_m\times K}$	$p(\boldsymbol{x}^{(m)}\vert\boldsymbol{h}) = \mathcal{MU}(\boldsymbol{x}^{(m)}\vert\hat{\boldsymbol{p}}^{(m)}),\ \hat{a}_c^{(m)} = a_c^{(m)} + \boldsymbol{W}_{c,:}^{(m)}\boldsymbol{h},\ \hat{\boldsymbol{p}}^{(m)} = [\frac{\exp(\hat{a}_1^{(m)})}{\sum_{c'=1}^{C_m}\exp(\hat{a}_{c'}^{(m)})}, \cdots, \frac{\exp(\hat{a}_{C_m}^{(m)})}{\sum_{c'=1}^{C_m}\exp(\hat{a}_{c'}^{(m)})}]^{\mathrm{T}}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + \sum_{m=1}^{M}(\boldsymbol{W}_{:,k}^{(m)})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}^{(m)}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n^{(m)}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s^{(m)}\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}^{(m)}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n^{(m)}\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s^{(m)}\boldsymbol{h}_s^{\mathrm{T}}\right)$
Multinomial-Bernoulli	$E(\boldsymbol{x},\boldsymbol{h}) = -\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \log\frac{n!}{\prod_{m=1}^{M}x_m!} - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{h} - \boldsymbol{x}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{h}$, where $\boldsymbol{a} = [\log p_1 + \log C, \cdots, \log p_M + \log C]^{\mathrm{T}}$, $\boldsymbol{b} = [\log\frac{p_1}{1-p_1}, \cdots, \log\frac{p_K}{1-p_K}]^{\mathrm{T}}$	$p(\boldsymbol{x}\vert\boldsymbol{h}) = \mathcal{MN}(\boldsymbol{x}\vert n, \hat{\boldsymbol{p}}),\ \hat{a}_m = a_m + \boldsymbol{W}_{m,:}\boldsymbol{h},\ \hat{p}_m = \frac{e^{\hat{a}_m}}{\sum_{m'=1}^{M}e^{\hat{a}_{m'}}}$ $p(\boldsymbol{h}\vert\boldsymbol{x}) = \prod_{k=1}^{K}\mathcal{BE}(\sigma(\hat{b}_k)),\ \hat{b}_k = b_k + (\boldsymbol{W}_{:,k})^{\mathrm{T}}\boldsymbol{x}$	$\triangle_{\boldsymbol{a}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\right)$ $\triangle_{\boldsymbol{b}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{h}_n\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{h}_s\right)$ $\triangle_{\boldsymbol{W}} \approx \frac{1}{N}\sum_{n=1}^{N}\left(-\boldsymbol{x}_n\boldsymbol{h}_n^{\mathrm{T}}\right) - \frac{1}{S}\sum_{s=1}^{S}\left(-\boldsymbol{x}_s\boldsymbol{h}_s^{\mathrm{T}}\right)$

generalized AIS procedure. The log-likelihood in Eq. (18) are still applicable to exp-RBM. To compute the log-likelihood of exp-RBM, we need to derive its free energy function $F(\boldsymbol{x})$ and estimate its log-partition function $\log Z$. Below, we present that both $F(\boldsymbol{x})$ and $\log Z$ can be beautifully computed.

$E(\boldsymbol{x}, \boldsymbol{h})$ in exp-RBMs is also decomposed in the form:

$$E(\boldsymbol{x}, \boldsymbol{h}) = -\zeta(\boldsymbol{x}) + \sum_{k=1}^{K} \gamma_k(\boldsymbol{x}, h_k). \tag{27}$$

Thus, the free energy function can be obtained analytically:

$$F(\boldsymbol{x}) = -\zeta(\boldsymbol{x}) - \sum_{k=1}^{K} \log \sum\!\!\!\!\!\!\int_{h_k} e^{-\gamma_k(\boldsymbol{x}, h_k)} = -\zeta(\boldsymbol{x}) - \sum_{k=1}^{K} B_k(\hat{\boldsymbol{b}}_k), \tag{28}$$

where $\hat{\boldsymbol{b}}_k$ and $B_k(\cdot)$ are respectively the posterior parameter and log-partition function of a hidden variable distribution. Readers should be able to distinguish the log-partition function $\log Z$ in exp-RBM from the log-partition functions $A_m(\cdot)$ and $B_k(\cdot)$ in exponential family distributions over visible and hidden variables. Examples of $\hat{\boldsymbol{b}}_k$ are defined in Table II. The log-partition functions $B_k(\cdot)$ of a variety of exponential family distributions are defined in Table I. Eq. (28) plays an important role in the estimation of log-partition functions of exp-RBMs and their deep extensions. $F(\boldsymbol{x})$ for selected exp-RBMs are derived and given in Table III.

Once defined the free energy function in the general form, we can describe the generalized AIS procedure, where the intermediate distribution for annealing can be defined as

$$p_t(\boldsymbol{x}, \boldsymbol{h}) = \frac{1}{Z_t} e^{(1-\beta_t)E_A(\boldsymbol{x}, \boldsymbol{h}^{(A)}) - \beta_t E_B(\boldsymbol{x}, \boldsymbol{h}^{(B)})}, \tag{29}$$

where the inverse temperature β_t gradually changes from 0 to 1: $0 = \beta_0 < \cdots < \beta_t < \cdots < \beta_T = 1$, so that when $\beta_0 = 0$, $p_0(\boldsymbol{x}, \boldsymbol{h}) = p_A(\boldsymbol{x}, \boldsymbol{h}^{(A)})$, i.e. the base-rate model A; when $\beta_T = 1$, $p_T(\boldsymbol{x}, \boldsymbol{h}) = p_B(\boldsymbol{x}, \boldsymbol{h}^{(B)})$, i.e. the focused model B. Using Eq. (28) and summing out the hidden variables $\boldsymbol{h}^{(A)}$ and $\boldsymbol{h}^{(B)}$, we have the intermediate marginals:

$$p_t(\boldsymbol{x}) = \frac{1}{Z_t} e^{-F_t(\boldsymbol{x})} = \frac{1}{Z_t} p_t^*(\boldsymbol{x}), \tag{30}$$

where $p_t^*(\boldsymbol{x}) = e^{-F_t(\boldsymbol{x})}$ and the free energy function is

$$\begin{aligned} F_t(\boldsymbol{x}) = &-(1-\beta_t)\zeta^{(A)}(\boldsymbol{x}) - \sum_{k=1}^{K_A} B_k\big((1-\beta_t)\hat{\boldsymbol{b}}_k^{(A)}\big) \\ &- \beta_t \zeta^{(B)}(\boldsymbol{x}) - \sum_{k=1}^{K_B} B_k\big(\beta_t \hat{\boldsymbol{b}}_k^{(B)}\big). \end{aligned} \tag{31}$$

To build the base-rate exp-RBM A, we assume, if model B is $\text{RBM}_B(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{W})$, model A is defined as $\text{RBM}_A(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{0})$. Removing the interaction term in Eq. (23), the partition function of model A can be analytically computed as

$$\begin{aligned} Z_A &= \sum\!\!\!\!\!\!\int_{\boldsymbol{x}} \sum\!\!\!\!\!\!\int_{\boldsymbol{h}} e^{\sum_{m=1}^{M}\left(\boldsymbol{a}_m^{\mathrm{T}} \boldsymbol{s}_m + \log f_m(x_m)\right) + \sum_{k=1}^{K}\left(\boldsymbol{b}_k^{\mathrm{T}} \boldsymbol{t}_k + \log g_k(h_k)\right)} \\ &= \prod_{m=1}^{M} \sum\!\!\!\!\!\!\int_{x_m} e^{\boldsymbol{a}_m^{\mathrm{T}} \boldsymbol{s}_m + \log f_m(x_m)} \prod_{k=1}^{K} \sum\!\!\!\!\!\!\int_{h_k} e^{\boldsymbol{b}_k^{\mathrm{T}} \boldsymbol{t}_k + \log g_k(h_k)} \qquad (32) \\ &= \prod_{m=1}^{M} e^{A_m(\boldsymbol{a}_m)} \prod_{k=1}^{K} e^{B_k(\boldsymbol{b}_k)}. \qquad (33) \end{aligned}$$

One can easily get stuck in Eq. (32), if attempting to enumerate all values of a variable as in the traditional binary RBM which has only two states for a variable. Thanks to the generalized definition of energy function [Eq. (23)], Z_A can be analytically computed by applying the concept of exponential family's log-partition function in Eq. (33)! Here, $A_m(\boldsymbol{a}_m)$ and $B_k(\boldsymbol{b}_k)$ are the log-partition functions of x_m and h_k, respectively. From Eq. (33), the log-partition function of the base-rate model A can be easily obtained:

$$\log Z_A = \sum_{m=1}^{M} A_m(\boldsymbol{a}_m) + \sum_{k=1}^{K} B_k(\boldsymbol{b}_k). \tag{34}$$

Exact forms of $\log Z_A$ are listed in Table III.

The Markov transition $T(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ can be defined as

$$p_t\big((x_t)_m|\boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}\big) = p_{\text{vis}}\big(x|\boldsymbol{\eta}((1-\beta_t)\boldsymbol{a}_m + \beta_t \hat{\boldsymbol{a}}_m)\big) \tag{35}$$

$$p_t\big((h_t^{(A)})_k|\boldsymbol{x}_t\big) = p_{\text{hid}}\big(h|\boldsymbol{\eta}((1-\beta_t)\boldsymbol{b}_k)\big) \tag{36}$$

$$p_t\big((h_t^{(B)})_k|\boldsymbol{x}_t\big) = p_{\text{hid}}\big(h|\boldsymbol{\eta}(\beta_t \hat{\boldsymbol{b}}_k)\big), \tag{37}$$

where function $\boldsymbol{\eta}(\cdot)$ converts the natural parameter to the standard forms, $\hat{\boldsymbol{a}}_m$ is computed as a combination of bias $\boldsymbol{a}_m$ and interaction with $\boldsymbol{h}_{t-1}^{(B)}$, $\hat{\boldsymbol{b}}_k$ is a combination of $\boldsymbol{b}_k$ and interaction with $\boldsymbol{x}_t$. Then, we can sample the sequence: $\{\boldsymbol{x}_t, \boldsymbol{h}_t^{(A)}, \boldsymbol{h}_t^{(B)}\}$ ($0 \le t \le T$-1), where $\boldsymbol{x}_t \sim p_t(\boldsymbol{x}_t|\boldsymbol{h}_{t-1})$, $\boldsymbol{h}_t^{(A)} \sim p_t(\boldsymbol{h}_t^{(A)}|\boldsymbol{x}_t)$, and $\boldsymbol{h}_t^{(B)} \sim p_t(\boldsymbol{h}_t^{(B)}|\boldsymbol{x}_t)$. Using this sequence, the importance weight can be computed by

$$w_s = \frac{p_1^*(\boldsymbol{x}_0)}{p_0^*(\boldsymbol{x}_0)} \cdots \frac{p_T^*(\boldsymbol{x}_{T-1})}{p_{T-1}^*(\boldsymbol{x}_{T-1})} = \prod_{t=0}^{T-1} \frac{p_{t+1}^*(\boldsymbol{x}_t)}{p_t^*(\boldsymbol{x}_t)}. \tag{38}$$

Then, the ratio of partition function can be estimated by

$$\frac{Z_B}{Z_A} \approx \frac{1}{S} \sum_{s=1}^{S} w_s = \hat{r}_{\text{AIS}}, \tag{39}$$

where Z_A can be analytically solved using Eq. (33) and S is the number of independent runs of AIS. When Z_A is computed using Eq. (33), we can estimate Z_B by

$$\hat{Z}_B = Z_A \times \hat{r}_{\text{AIS}}. \tag{40}$$

As suggested in [18], when the number of hidden units are extremely large, we can instead consider a stable version of AIS by taking logarithm on both sides of Eq. (40):

$$\log \hat{Z}_B = \log \hat{r}_{\text{AIS}} + \log Z_A, \tag{41}$$

Since the log-importance weight is

$$\begin{aligned} l_s = \log w_s &= \sum_{t=0}^{T-1} \big(\log p_{t+1}^*(\boldsymbol{x}_t) - \log p_t^*(\boldsymbol{x}_t)\big) \\ &= \sum_{t=0}^{T-1} \big(F_t(\boldsymbol{x}_t) - F_{t+1}(\boldsymbol{x}_t)\big), \end{aligned} \tag{42}$$

thus, $\hat{r}_{\text{AIS}} = \frac{1}{S}\sum_{s=1}^{S} e^{l_s}$. Then, we obtain

$$\log \hat{Z}_B = \log \frac{1}{S} \sum_{s=1}^{S} e^{l_s} + \log Z_A. \tag{43}$$

TABLE III: Estimatation of log-partition functions of exp-RBMs.

Model	$F(\boldsymbol{x})$	$F_t(\boldsymbol{x})$ in $p_t(\boldsymbol{x})$	$T(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$	$\log Z_A$
Bernoulli-Bernoulli	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{BE}(x_m \mid \sigma((1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1}^{(B)})))$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M} \log(1 + e^{a_m}) + \sum_{k=1}^{K} \log(1 + e^{b_k})$
Gaussian-Bernoulli	$-\boldsymbol{a}^{(1)\mathrm{T}}\boldsymbol{x} - \boldsymbol{a}^{(2)\mathrm{T}}\boldsymbol{x}^{*2} - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{(1)\mathrm{T}}\boldsymbol{x} - \boldsymbol{a}^{(2)\mathrm{T}}\boldsymbol{x}^{*2} - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{GS}\Big((x_t)_m \mid -\frac{(1-\beta_t)a_m^{(1)} + \beta_t(a_m^{(1)} + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1}^{(B)})}{a_m^{(2)}}, (-2a_m^{(2)})^{-1}\Big)$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M} (\frac{1}{2}\log\frac{\pi}{-a_m^{(2)}} - \frac{a_m^{(1)\,2}}{4a_m^{(2)}}) + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Gaussian-Bernoulli fix precision, model 1	$-\boldsymbol{a}^{\mathrm{T}}(\boldsymbol{\lambda} * \boldsymbol{x}) - \sum_{m=1}^{M}\Big(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2\Big) - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}(\boldsymbol{\lambda}*\boldsymbol{x})})$	$-\boldsymbol{a}^{\mathrm{T}}(\boldsymbol{\lambda} * \boldsymbol{x}) - \sum_{m=1}^{M}\Big(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2\Big) - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}(\boldsymbol{\lambda}*\boldsymbol{x}))})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{GS}\Big((x_t)_m \mid (1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1}^{(B)}), \lambda_m\Big)$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}(\boldsymbol{\lambda} * \boldsymbol{x}_t))))$	$\sum_{m=1}^{M}(\frac{\lambda_m}{2}a_m^2) + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Gaussian-Bernoulli fix precision, model 2	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}\Big(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2\Big) - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}\Big(\log\frac{\sqrt{\lambda_m}}{\sqrt{2\pi}} - \frac{\lambda_m}{2}x_m^2\Big) - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{GS}\Big((x_t)_m \mid \frac{(1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1}^{(B)})}{\lambda}, \lambda_m\Big)$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M}(\frac{a_m^2}{2\lambda_m}) + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Poisson-Bernoulli	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{PO}((x_t)_m \mid e^{(1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1})})$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M} e^{a_m} + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Poisson-Binomial	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \sum_{k=1}^{K} n_k \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} + \sum_{m=1}^{M}\log(x_m!) - \sum_{k=1}^{K} \Big(n_k \log(1 + e^{(1-\beta_t)b_k}) + n_k \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{PO}((x_t)_m \mid e^{(1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1})})$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BN}((h_t^{(A)})_k \mid n_k, \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BN}((h_t^{(B)})_k \mid n_k, \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M} e^{a_m} + \sum_{k=1}^{K} n_k \log(1 + e^{b_k})$
Negative-Binomial-Bernoulli	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}\log\frac{(x_m + s_m - 1)!)}{(s_m - 1)!x_m!} - \sum_{k=1}^{K} \log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \sum_{m=1}^{M}\log\frac{(x_m + s_m - 1)!)}{(s_m - 1)!x_m!} - \sum_{k=1}^{K} \Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t((x_t)_m \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{NB}((x_t)_m \mid s_m, 1 - e^{(1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1})})$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$\sum_{m=1}^{M} -s_m \log(1 - e^{a_m}) + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Multinoulli-Bernoulli	$-\sum_{m=1}^{M}\boldsymbol{a}^{(m)\mathrm{T}}\boldsymbol{x}^{(m)} - \sum_{k=1}^{K}\log(1 + e^{b_k + \sum_{m=1}^{M}(\boldsymbol{W}^{(m)\mathrm{T}})_{k,:}\boldsymbol{x}^{(m)}})$	$-\sum_{m=1}^{M}\boldsymbol{a}^{(m)\mathrm{T}}\boldsymbol{x}^{(m)} - \sum_{k=1}^{K}\Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \sum_{m=1}^{M}(\boldsymbol{W}^{(m)\mathrm{T}})_{k,:}\boldsymbol{x}^{(m)})})\Big)$	$p_t(\boldsymbol{x}_t^{(m)} \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{MU}\Big(\boldsymbol{x}_t^{(m)} \mid \big[\frac{e^{(1-\beta_t)a_c^{(m)} + \beta_t(a_c^{(m)} + \boldsymbol{W}_{c,:}^{(m)}\boldsymbol{h}_{t-1})}}{\sum_{c'=1}^{C_m} e^{(1-\beta_t)a_{c'}^{(m)} + \beta_t(a_{c'}^{(m)} + \boldsymbol{W}_{c',:}^{(m)}\boldsymbol{h}_{t-1})}}\big]\Big)$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \sum_{m=1}^{M}(\boldsymbol{W}^{(m)\mathrm{T}})_{k,:}\boldsymbol{x}_t^{(m)})))$	$\sum_{m=1}^{M}\log\sum_{c'=1}^{C_m} e^{a_{c'}^{(m)}} + \sum_{k=1}^{K}\log(1 + e^{b_k})$
Multinomial-Bernoulli	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \log\frac{n!}{\prod_{m=1}^{M} x_m!} - \sum_{k=1}^{K}\log(1 + e^{b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}})$	$-\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} - \log\frac{n!}{\prod_{m=1}^{M} x_m!} - \sum_{k=1}^{K}\Big(\log(1 + e^{(1-\beta_t)b_k}) + \log(1 + e^{\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x})})\Big)$	$p_t(\boldsymbol{x}_t \mid \boldsymbol{h}_{t-1}^{(A)}, \boldsymbol{h}_{t-1}^{(B)}) = \mathcal{MN}\Big(\boldsymbol{x}_t \mid n, \big[\frac{e^{(1-\beta_t)a_m + \beta_t(a_m + \boldsymbol{W}_{m,:}\boldsymbol{h}_{t-1})}}{\sum_{m'=1}^{M} e^{(1-\beta_t)a_{m'} + \beta_t(a_{m'} + \boldsymbol{W}_{m',:}\boldsymbol{h}_{t-1})}}\big]\Big)$ $p_t((h_t^{(A)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(A)})_k \mid \sigma((1-\beta_t)b_k))$ $p_t((h_t^{(B)})_k \mid \boldsymbol{x}_t) = \mathcal{BE}((h_t^{(B)})_k \mid \sigma(\beta_t(b_k + \boldsymbol{W}^{\mathrm{T}}{}_{k,:}\boldsymbol{x}_t)))$	$n\log\sum_{m=1}^{M} e^{a_m} + \sum_{k=1}^{K}\log(1 + e^{b_k})$

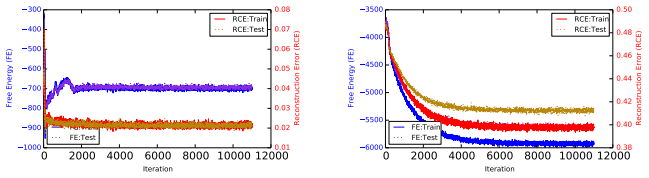
The free energy function, intermediate distribution, Markov transition function and base-rate model A's log-partition function for instances of exp-RBMs are summarized in Table III.

IV. Experiments

We demonstrate the performance of exp-RBMs using the `20 Newsgroups` data [19], which has 11,269 training and 7,505 test documents. They were converted to word counts. Stop-words and words used in less than 60 training samples were removed, leaving 4,031 words for analysis.

A. Free Energy, Partition Function and Likelihood

We ran Bernoulli-Bernoulli, Gaussian-Bernoulli, Poisson-Bernoulli, Negative-Binomial-Bernoulli and Multinomial-Bernoulli RBMs on the training set to track changes of *mean reconstruction error* and *mean free energy* along with the evolution of learning on the training and test sets. Both measures are recommended in [15] to monitor the learning of RBM. Data for Bernoulli visible type were binarized. Data for Gaussian visible type were normalized with tf-idf. And each sample for Poisson, negative-binomial and multinomial models was normalized to have 1,000 counts. The number of hidden units for each model is given in Table IV. For all these five models, the reconstruction error rapidly decreases and then plateaus, implying that the models learned knowledge from the data, where the reconstruction error acts as a good indicator to monitor the learning progress of exp-RBMs (Figure 1). Interestingly, the free energy does not necessarily decrease in learning. We notice that the free energy function of Bernoulli-Bernoulli RBM may stay flat for some parameter settings (Figure 1a), though it usually has behaviour similar to that under the reconstruction error, as depicted in Figure 1b.



(a) Bernoulli-Bernoulli (b) Multinomial-Bernoulli

Fig. 1: Change of free energy and reconstruction error.

TABLE IV: Free energy and estimated log-partition function and log-likelihood of selected exp-RBMs.

Vis. Type (# Hid.)	Data	$F(\boldsymbol{x})$	$\log Z$ (STD)	$\log p(\boldsymbol{x})$
Bernoulli(1000)	Train	-699.06	903.70 ($\pm$1.74)	-204.64
	Test	-693.87	903.81 ($\pm$0.79)	-209.94
Gaussian(500)	Train	-7178.64	348.67 ($\pm$0.03)	6829.96
	Test	-7178.82	348.66 ($\pm$0.03)	6830.16
Poisson(500)	Train	1729.99	1148.25 ($\pm$3.15)	-2878.25
	Test	2061.52	1147.73 ($\pm$5.16)	-3209.25
Neg. Bin.(500)	Train	1309.17	124.54 ($\pm$1.12)	-1433.71
	Test	1311.64	124.64 ($\pm$1.40)	-1436.28
Multinomial(500)	Train	-5927.48	8818.54 ($\pm$1.49)	-2891.05
	Test	-5622.90	8818.56 ($\pm$2.37)	-3195.65

After learning, we computed their mean free energy using Eq. (28), estimated their log-partition functions using our generalized AIS method, and calculated their log-likelihoods on both training and test sets (Table IV). First, from the definitions of Poisson-Bernoulli and Negative-Binomial-Bernoulli RBMs' free energy functions in Table III, we learned that their values could be positive due to large values of the $\zeta(\boldsymbol{x})$ function, which is corroborated by our results in Table IV. Second, the log-partition function were estimated with small variances, indicating that the generalized AIS procedure worked stably as the original AIS for binary RBM. Third, the log-likelihood of all these models are close to zero (except the Gaussian model because the likelihoods or densities of continuous distributions could be arbitrarily high), showing a good learning quality. Note that the models learned on different normalized data cannot be compared. However, we can compare Poisson-Bernoulli, Negative-Binomial-Bernoulli and Multinomial-Bernoulli RBMs as they were trained on the same data. We can observe that Poisson-Bernoulli and Multinomial-Bernoulli RBMs obtained similar log-likelihood, while the Negative-Binomial-Bernoulli RBM gained a better fitting. We generated samples from each the learned models using a long run of Gibbs sampling, with some shown in Table V. All models generated meaningful word profiles.

B. Associative Sampling and Classification

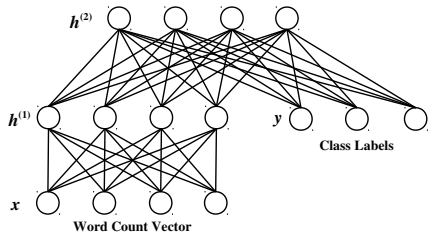


Fig. 2: A two-view deep Boltzmann machine.

There are three strategies to apply exp-RBM to a classification problem. The first method is to use RBM as a feature extraction method and employ any classifier on the generated new features. The second strategy is to adapt a generative model to be a discriminative model by transforming generative connections to recognition connections, adding softmax units over the top hidden layer, and fine-tuning the parameters. The third is to design a two-view deep generative model by adding a class view. This strategy is first used in DBN for handwritten digits classification [5]. We are particularly interested in the third strategy as such a model can (1) generate both novel samples and the corresponding annotations, (2) generate class-specific samples when the class label is given as attention, and (3) predict the class label with probability. To demonstrate these potentials of the exp-RBMs, we designed a two-view deep Boltzmann machine (DBM) with 500 units in the first and second hidden layers, respectively (see Figure 2, sophisticated deep treatment of exp-RBMs will be reported separately). The word count view is assumed to follow multinomial or Poisson

TABLE V: Top ten words in samples generated by exp-RBM.

Bernoulli	writes	space	nasa	article	earth	high	money	cost	orbit	gov
	mail	send	message	email	place	university	give	address	set	post
	god	jesus	christ	christians	bible	writes	rutgers	people	man	church
Multinomial	pc	mail	card	controller	cable	ibm	shipping	modem	monitor	price
	gun	people	article	don	writes	time	control	crime	weapons	apr
	god	jesus	christ	lord	writes	church	john	heaven	bible	father

TABLE VI: Samples generated by the two-view DBM.

Sampling Method	$\boldsymbol{x}$										y
Sample $\boldsymbol{x}$ and y	space	billion	moon	nasa	launch	cost	shuttle	year	program	news	sci.space
	unix	system	software	dos	support	os	graphics	user	internet	manager	comp.windows.x
	team	don	play	games	runs	win	center	game	guy	give	rec.sport.baseball
Clamp y, sample $\boldsymbol{x}$	god	people	writes	exist	article	existence	atheism	belief	question	moral	alt.atheism
	memory	time	file	system	running	mb	dos	ram	swap	read	comp.sys.ibm.pc.hardware
	advance	price	software	pc	windows	dos	mail	package	ms	programs	misc.forsale

distributions, while the class view follows a multinoulli distribution. This two-view DBM was trained based on mean-field approximation and Gibbs sampling [16]. When sampling both views using this learned model, we applied Gibbs sampling. We found the generated word profiles and the corresponding class labels were well associated. Some examples of the generated samples are given in Table VI. When one view is clamped to generate samples of the other view, we employed the mean-field variational approximation method. Since the class view has a small influence on the associative memory, we found that a large portion of the generated word profiles given a class label fixed may not be related to the specified class. Finally, when we used multinomial or Poisson types and clamped the word profile in test, this two-view DBM was able to predict its class label with an accuracy of 63.57% or 68.35%. We also ran a multilayer perceptrons with two hidden layers (each has 500 RELU units) as a discriminative model and obtained an accuracy of 58.53%. Note that our main objective here is demonstrating the versatile potentials of exp-RBMs in deep generative models, but not competing with a discriminative classifier.

V. Conclusions

In this paper, we revisit the concepts of exponential family RBMs and formulate the generalized free energy function and annealed importance sampling to estimate their log-partition functions. We demonstrate in a text processing task that the five instances of exp-RBMs can capture knowledge from the data and the estimated log-partition functions are stable. Using a two-view deep Boltzmann machine based on exp-RBMs, we demonstrate promising potentials of exp-RBMs in development of deep generative models. As our future work, we will design a variety of deep generative models as well as (stochastic and deterministic) convolutional and recurrent networks based on the theory presented in this paper. We will make our implementation of exp-RBMs publicly available.

Acknowledgment

This research was supported by the National Research Council Canada. We thank the reviewers for their thoughtful comments.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[2] R. Neal, "Connectionist learning of belief networks," *Artificial Intelligence*, vol. 56, pp. 71–113, 1992.

[3] P. Dayan, G. Hinton, R. Neal, and R. Zemel, "The Helmholtz machine," *Neural Computation*, vol. 7, pp. 1022–1037, 1995.

[4] P. Smolensky, "Information processing in dynamical systems: Foundations of harmony theory," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundation*, D. Rumelhart and J. McLelland, Eds. Cambridge, MA: MIT, 1986, ch. 6, pp. 194–281.

[5] G. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

[6] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machine," in *International Conference on Artificial Intelligence and Statistics*, 2009, pp. 448–455.

[7] Y. Bengio, "Learning deep architectures for AI," *Foundations and Trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[8] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504–507, 2006.

[9] R. Salakhutdinov and G. Hinton, "Replicated softmax: An undirected topic model," in *Advances in Neural Information Processing Systems*, 2009, pp. 1607–1614.

[10] M. Welling, M. Rosen-zvi, and G. Hinton, "Exponential family harmoniums with an application to information retrieval," in *Advances in Neural Information Processing Systems*, 2005, pp. 1481–1488.

[11] G. Hinton and T. Sejnowski, "Optimal perceptual inference," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1983, pp. 448–453.

[12] G. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[13] T. Tieleman, "Training restricted Boltzmann machines using approximations to the likelihood gradient," in *International Conference on Machine Learning*, 2008, pp. 1064–1071.

[14] R. Salakhutdinov and I. Murray, "On the quantitative analysis of deep belief networks," in *International Conference on Machine Learning*, 2008, pp. 872–879.

[15] G. Hinton, "A practical guide to training restricted Boltzmann machines," Department of Computer Science, University of Toronto, Toronto, Ontario, Canada, Tech. Rep., 2010.

[16] N. Srivastava and R. Salakhutdinov, "Multimodal learning with deep Boltzmann machines," *Journal of Machine Learning Research*, vol. 15, pp. 2949–2980, 2014.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[18] R. Neal, "Annealed importance sampling," *Statistics and Computing*, vol. 11, pp. 125–139, 2001.

[19] K. Lang, "Newsweeder: Learning to filter netnews," in *International Conference on Machine Learning*, 1995, pp. 331–339, http://qwone.com/ jason/20Newsgroups.