Gamma Boltzmann Machine for Audio Modeling

Toru Nakashika, *Member, IEEE*, and Kohei Yatabe, *Member, IEEE*

***Abstract*—This paper presents an energy-based probabilistic model that handles nonnegative data in consideration of both linear and logarithmic scales. In audio applications, magnitude of time-frequency representation, including spectrogram, is regarded as one of the most important features. Such magnitude-based features have been extensively utilized in learning-based audio processing. Since a logarithmic scale is important in terms of auditory perception, the features are usually computed with a logarithmic function. That is, a logarithmic function is applied within the computation of features so that a learning machine does not have to explicitly model the logarithmic scale. We think in a different way and propose a restricted Boltzmann machine (RBM) that simultaneously models linear- and log-magnitude spectra. RBM is a stochastic neural network that can discover data representations without supervision. To manage both linear and logarithmic scales, we define an energy function based on both scales. This energy function results in a conditional distribution (of the observable data, given hidden units) that is written as the gamma distribution, and hence the proposed RBM is termed *gamma-Bernoulli RBM*. The proposed gamma-Bernoulli RBM was compared to the ordinary Gaussian-Bernoulli RBM by speech representation experiments. Both objective and subjective evaluations illustrated the advantage of the proposed model.**

***Index Terms*—Boltzmann machine, nonnegative data modeling, gamma distribution, speech parameterization, speech synthesis.**

I. Introduction

LEARNING data representation is a fundamental task, and many methods have been proposed, e.g., variational autoencoders (VAEs) [1]–[3], generative adversarial networks (GANs) [4]–[7], autoregressive (AR) models [8], [9], and normalizing flows [10], [11]. One theoretically well-founded model for this task is the Boltzmann machine [12]. It is a stochastic neural network that can automatically discover data representations in terms of probability distribution. Some advantages of the Boltzmann machine include its interpretability as a generative model with good prospect, and the number of parameters for training is relatively small. The restricted Boltzmann machine (RBM) [13] is a computationally efficient variant of Boltzmann machines. Since RBMs can be trained with computational effort less than the other models, RBMs has been successfully utilized in various applications involving pattern recognition and machine learning, e.g., computer vision [14], collaborative filtering [15], and even geochemical analysis [16], to name a few. Its capability in discovering latent representations without supervision has potential for further expansion, and hence it should be worthwhile to develop an RBM for a specific class of data.

In audio applications, the Gaussian-Bernoulli RBM [17], [18] has been utilized for modeling signals through the magnitude spectra. Since spectral components are important for auditory perception, audio signals can be well characterized in the frequency domain. Therefore, RBMs are trained to approximate the probability distribution of the given data in a domain related to frequency. For example, many studies have applied RBMs to model the mel-frequency cepstral coefficients (MFCC) [19], [20] or mel-cepstral features [21], [22] of speech signals. For extracting richer information from the signals, raw magnitude or STRAIGHT [23] spectra have also been considered [24]–[26]. Moreover, some studies attempted modeling the raw signals using RBMs [27], [28]. These representations are real-valued, and hence a Gaussian-Bernoulli RBM is a natural choice for audio modeling because it handles the observable data through the Gaussian distribution (as opposed to the original RBM defined for binary signals). Since magnitude spectrum is the standard representation of audio signals, this paper focuses on modeling of magnitude spectra.

For modeling magnitude spectra, their two aspects must be carefully taken into account: nonnegativity and logarithmic scale. Firstly, magnitude spectra are essentially nonnegative-valued. Since calculation of magnitude spectra involves absolute value, by definition, negative values never appear in the observable data to be modeled. Erroneous negative sign results in 180° phase shift, and hence nonnegativity of the modeled magnitude spectra must be maintained. Secondly, modeled magnitude spectra should be accurate in terms of a logarithmic scale. The human auditory system perceives magnitude of sound in the logarithmic-like scale rather than a linear scale. Based on this fact, many handcrafted audio features as MFCC involves the logarithmic operation within their calculation processes. A model of magnitude spectra must handle the data logarithmically in a reasonable manner.

However, a Gaussian-Bernoulli RBM has trouble in considering the above two aims. The Gaussian distribution allows negative values that are not consistent with the concept of magnitude. It is not straightforward to limit the Gaussian distribution into nonnegative values, and therefore the learned representation should contain unavoidable model error. Taking logarithm of data before inputting to an RBM may seem a solution to this problem. Nevertheless, the asymmetric nature of the logarithmic

Manuscript received September 8, 2020; revised April 11, 2021 and June 17, 2021; accepted July 4, 2021. Date of publication July 8, 2021; date of current version August 13, 2021. This work was supported by JSPS KAKENHI under Grant 21K11957. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yu Tsao. *(Corresponding author: Toru Nakashika).*

Toru Nakashika is with the Graduate School of Informatics and Engineering, The University of Electro-Communications, Tokyo 182-8585, Japan (e-mail: nakashika@uec.ac.jp).

Kohei Yatabe is with the Department of Intermedia Art and Science, Waseda University, Tokyo 169-8555, Japan (e-mail: k.yatabe@asagi.waseda.jp).

Digital Object Identifier 10.1109/TASLP.2021.3095656

function can make the training difficult for symmetric models as the Gaussian distribution. Moreover, log-magnitude of approximately sparse spectra (e.g., those of typical audio and speech signals) can cause extreme outliers when the magnitude is around zero. Thus, a specific mechanism for audio modeling must be developed.

In this paper, we propose a variant of RBMs called *gamma-Bernoulli RBM* for modeling magnitude spectra in consideration of the logarithmic scale.[1] To manage both linear and logarithmic scales, we define an energy function consisting of the usual quadratic term and an additional log-magnitude term. This energy function provides a general gamma Boltzmann machine that simultaneously considers linear- and log-magnitude spectra. The term *gamma* is assigned to this Boltzmann machine because its conditional distribution of a unit (given the other units) is the gamma distribution. Then, its connection is restricted to form the gamma-Bernoulli RBM. The proposed RBM represents the conditional distribution of the visible units (given hidden units) by the gamma distribution, which naturally limits the domain of data to positive numbers. We also propose several variants of the gamma-Bernoulli RBM by considering combinations of the trainable parameters. The optimal model among the proposed RBMs was investigated by speech representation experiments. Both objective and subjective evaluations illustrated the advantage of the gamma-Bernoulli RBM.

The rest of the paper is organized as follows. In Section II, the ordinary Boltzmann machine and RBMs are summarized for contrasting the difference between the conventional and proposed models. Then, the proposed models are discribed in Section III. Specifically, a general gamma Boltzmann machine is introduced in Section III-A. Furthermore, by restricting its connection, the gamma-Bernoulli RBM is proposed in Section III-C. Some notes on implementation and optimization of the gamma-Bernoulli RBM follow them. The properties and performances of the proposed RBMs are experimentally investigated in Section IV. An extension of the gamma-Bernoulli RBM and a gamma-gamma RBM are additionally proposed in Appendices A and B, respectively, for experimental investigations. Finally, the paper is concluded in Section V.

II. Preliminaries

A. Boltzmann Machine

The Boltzmann machine [12] is an unsupervised neural network for approximating a probability distribution of a set of given data. Let $\mathcal{X}$ be a space of the variables under investigation, and its element be denoted by $\boldsymbol{x} \in \mathcal{X}$ (both $\boldsymbol{x}$ and $\mathcal{X}$ will be clarified later). Then, a Boltzmann machine represents a probability density function (PDF) of $\boldsymbol{x}$ as

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

where $Z = \int_{\mathcal{X}} \mathrm{e}^{-E(\boldsymbol{x})}\,\mathrm{d}\boldsymbol{x}$ is the normalizing constant called partition function, and $E(\cdot)$ is the so-called energy function. A Boltzmann machine is defined through the energy function. In this section, the following energy function involving a matrix $\mathbf{U}$ and a vector $\mathbf{u}$ is considered for the conventional models:

$$E(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^{\mathsf{T}}\mathbf{U}\boldsymbol{x} - \mathbf{u}^{\mathsf{T}}\boldsymbol{x}, \tag{2}$$

where the explicit forms of $\mathbf{U}$ and $\mathbf{u}$ are given later.

B. Restricted Boltzmann Machine (RBM)

RBM is one of the most practical (and hence important) variants of a Boltzmann machine. The above general Boltzmann machine may not be practical because calculation (or even approximation) of the integral is difficult. For a practical dimensionality, its training can be extremely slow. To develop a fast training algorithm and avoid such difficulty, RBMs restrict the connection between the units.

An RBM separates the variables into two parts: the visible and hidden variables denoted by $\boldsymbol{v}$ and $\boldsymbol{h}$, respectively, where an element of these vectors is called *unit*. The vector $\boldsymbol{v}$ corresponds to observable data (and hence *visible*), while $\boldsymbol{h}$ represents the latent variables for conditional hidden representation of the data. That is, a PDF of the visible variable $\boldsymbol{v}$ is given by the following marginalization:

$$p(\boldsymbol{v}) = \int_{\mathcal{H}} p(\boldsymbol{v},\boldsymbol{h})\,\mathrm{d}\boldsymbol{h} = \frac{1}{Z}\int_{\mathcal{H}} \mathrm{e}^{-E(\boldsymbol{v},\boldsymbol{h})}\,\mathrm{d}\boldsymbol{h}, \tag{3}$$

where $Z = \int_{\mathcal{V}\times\mathcal{H}} \mathrm{e}^{-E(\boldsymbol{v},\boldsymbol{h})}\,\mathrm{d}\boldsymbol{v}\mathrm{d}\boldsymbol{h}$, and $\mathcal{V}$ and $\mathcal{H}$ are the spaces of visible and hidden variables, respectively.

The connections between units are determined by the energy function. An energy function of RBM $E(\boldsymbol{v},\boldsymbol{h})$ is defined such that both visible and hidden units do not have interconnections. In other words, RBM does not have visible-visible and hidden-hidden connections. This property is realized by setting all non-diagonal elements of a square matrix $\mathbf{W}$ of quadratic forms $\boldsymbol{v}^{\mathsf{T}}\mathbf{W}\boldsymbol{v}$ and $\boldsymbol{h}^{\mathsf{T}}\mathbf{W}\boldsymbol{h}$ to zero. Such restriction enables fast training by sampling from two conditional distributions: $p(\boldsymbol{v}|\boldsymbol{h})$ and $p(\boldsymbol{h}|\boldsymbol{v})$. These conditional probabilities are the key ingredients of RBMs and characterize the type of an RBM.

C. Bernoulli-Bernoulli RBM

The original RBM [13] was defined for binary variables, i.e., $\mathcal{V}$ and $\mathcal{H}$ are the sets of binary vectors: $\boldsymbol{v} \in \{0,1\}^D$, $\boldsymbol{h} \in \{0,1\}^H$, where D and H are the dimensions of $\boldsymbol{v}$ and $\boldsymbol{h}$, respectively. Its energy function $E_{\mathrm{BB}}(\boldsymbol{v},\boldsymbol{h})$ is defined as

$$E_{\mathrm{BB}}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{v}^{\mathsf{T}}\mathbf{W}\boldsymbol{h} - \boldsymbol{b}^{\mathsf{T}}\boldsymbol{v} - \boldsymbol{c}^{\mathsf{T}}\boldsymbol{h}, \tag{4}$$

that is related to the general Boltzmann machine in Eq. (2) as

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v}\\ \boldsymbol{h}\end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix}\mathbf{O} & \mathbf{W}\\ \mathbf{W}^{\mathsf{T}} & \mathbf{O}\end{bmatrix}, \quad \boldsymbol{u} = \begin{bmatrix}\boldsymbol{b}\\ \boldsymbol{c}\end{bmatrix}, \tag{5}$$

where $\mathbf{W} \in \mathbb{R}^{D\times H}$, $\boldsymbol{b} \in \mathbb{R}^D$, $\boldsymbol{c} \in \mathbb{R}^H$, $\mathbf{O}$ represents the all-zero matrix with appropriate size, and the operations between the binary and real numbers are performed by regarding the binary symbols as real numbers.

[1] A preliminary version of this study has been published in the proceedings of 12th APSIPA ASC [29]. In this paper, we intended to extend the preliminary study by proposing several variants of the gamma-Bernoulli RBM and experimentally investigating the optimal choice of the model. Both objective and subjective comparisons were conducted with deeper discussions.

An RBM defined by the above energy function $E_{\mathrm{BB}}(\boldsymbol{v},\boldsymbol{h})$ is called Bernoulli-Bernoulli RBM. This is because the two conditional probabilities required for its training result in the element-wise Bernoulli distributions $\mathcal{B}(\,\cdot\,;\boldsymbol{p})$:

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{B}(\boldsymbol{v}; f_\sigma[\,\boldsymbol{b}+\mathbf{W}\boldsymbol{h}\,]), \tag{6}$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \mathcal{B}(\boldsymbol{h}; f_\sigma[\,\boldsymbol{c}+\mathbf{W}^{\mathsf{T}}\boldsymbol{v}\,]), \tag{7}$$

where $\boldsymbol{p}\in[0,1]^D$ (or $[0,1]^H$) is a vector representing the probabilities of taking the value 1 for each element, and $f_\sigma[\,\cdot\,]$ denotes the element-wise sigmoid function.

D. Gaussian-Bernoulli RBM

The Bernoulli-Bernoulli RBM has a severe practical limitation. Even though many of the interesting real-world data are not binary in nature, it can only handle binary data. To avoid this limitation, the Gaussian-Bernoulli RBM [17] modified the definition of the above energy function as follows: [2]

$$E_{\mathrm{GB}}(\boldsymbol{v},\boldsymbol{h}) = \frac{1}{2}\boldsymbol{v}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{v} - \boldsymbol{v}^{\mathsf{T}}\mathbf{W}\boldsymbol{h} - \boldsymbol{b}^{\mathsf{T}}\boldsymbol{v} - \boldsymbol{c}^{\mathsf{T}}\boldsymbol{h}, \tag{8}$$

that is related to the general Boltzmann machine in Eq. (2) as

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v}\\ \boldsymbol{h}\end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix}-\boldsymbol{\Sigma}^{-1} & \mathbf{W}\\ \mathbf{W}^{\mathsf{T}} & \mathbf{O}\end{bmatrix}, \quad \boldsymbol{u} = \begin{bmatrix}\boldsymbol{b}\\ \boldsymbol{c}\end{bmatrix}, \tag{9}$$

where $\boldsymbol{\Sigma}=\mathrm{diag}(\boldsymbol{\sigma}^2)$ is a diagonal matrix, $\boldsymbol{\sigma}^2\in\mathbb{R}_{++}^D$ is the model parameter representing variance of the visible variables ($\mathbb{R}_{++}$ is the set of positive numbers), and $\mathrm{diag}(\cdot)$ is the operator constructing the diagonal matrix from an input vector. An RBM defined by this energy function $E_{\mathrm{GB}}(\boldsymbol{v},\boldsymbol{h})$ can naturally handle real-valued data $\boldsymbol{v}\in\mathbb{R}^D$, while the hidden variables are remained binary, $\boldsymbol{h}\in\{0,1\}^H$. Note that the difference of $E_{\mathrm{GB}}(\boldsymbol{v},\boldsymbol{h})$ from $E_{\mathrm{BB}}(\boldsymbol{v},\boldsymbol{h})$ in Eq. (4) is just the first term $\boldsymbol{v}^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{v}$ that represents the self-connection of the visible units. This term does not introduce interconnection of the visible units because the matrix $\boldsymbol{\Sigma}^{-1}$ does not have any non-diagonal element.

An RBM defined by the energy function in Eq. (8) is called Gaussian-Bernoulli RBM since its conditional probabilities are

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{N}(\boldsymbol{v}; \boldsymbol{\Sigma}(\boldsymbol{b}+\mathbf{W}\boldsymbol{h}), \boldsymbol{\Sigma}), \tag{10}$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \mathcal{B}(\boldsymbol{h}; f_\sigma[\,\boldsymbol{c}+\mathbf{W}^{\mathsf{T}}\boldsymbol{v}\,]), \tag{11}$$

where $\mathcal{N}(\,\cdot\,;\boldsymbol{\mu},\boldsymbol{\Sigma})$ is the Gaussian distribution with a mean vector $\boldsymbol{\mu}\in\mathbb{R}^D$ and a covariance matrix $\boldsymbol{\Sigma}\in\mathbb{R}^{D\times H}$. That is, observable data are handled by the Gaussian distribution. Hence, it can approximate the distribution of real-valued data by learning the parameters $(\boldsymbol{\Sigma},\mathbf{W},\boldsymbol{b},\boldsymbol{c})$ from the given data.

III. Gamma Boltzmann Machine

Among Boltzmann machines, the Gaussian-Bernoulli RBM has been utilized as a standard choice for real-world applications.

[2]Note that this definition is somewhat different from those defined in [17] or [18]. We defined the energy function as in Eq. (8) because we empirically found that this works better for our application in Section IV.

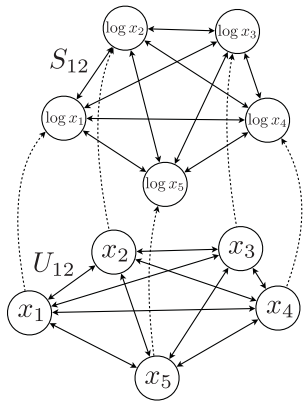

Fig. 1. Graphical representation of the proposed gamma Boltzmann machine, which handles the data not only in linear domain but also in logarithmic domain. U_{12} indicates the bidirectional-connection weight between x_1 and x_2, while S_{12} indicates the weight between $\log(x_1)$ and $\log(x_2)$.

This should be because it can naturally handle real-valued signals, and the Gaussian distribution is one of the most fundamental distributions in science and engineering. In audio applications, one essential and important target of generative modeling is magnitude spectrum [7], [26], [30]–[32]. However, as mentioned in the Introduction (4th paragraph), Gaussian-Bernoulli RBMs have two issues on modeling magnitude spectra: production of negative values and mismatch to the logarithmic scale. To avoid these issues, we propose a new variant of Boltzmann machines named *gamma-Bernoulli RBM*.

A. Proposed Gamma Boltzmann Machine

At first, we propose a general *gamma Boltzmann machine* without restriction and explain its relation to the gamma distribution. As summarized in the previous section, Boltzmann machines are characterized by the energy function. Since our purpose in this paper is to represent data in consideration of a logarithmic scale, we propose the following energy function:

$$E_\Gamma(\boldsymbol{x}) = -\frac{1}{2}\boldsymbol{x}^{\mathsf{T}}\mathbf{U}\boldsymbol{x} - \boldsymbol{u}^{\mathsf{T}}\boldsymbol{x} - \frac{1}{2}\log(\boldsymbol{x})^{\mathsf{T}}\mathbf{S}\log(\boldsymbol{x}) - \boldsymbol{s}^{\mathsf{T}}\log(\boldsymbol{x}), \tag{12}$$

as in Fig. 1, where $-\mathbf{U}\in\mathbb{R}_{++}^{I\times I}$, $\mathbf{U}^T=\mathbf{U}$, $U_{ii}=0\ \forall i$, $\mathbf{S}\in\mathbb{R}_{++}^{I\times I}$, $\mathbf{S}^T=\mathbf{S}$, $S_{ii}=0\ \forall i$, $-\boldsymbol{u}\in\mathbb{R}_+^I$, and $\boldsymbol{s}+1\in\mathbb{R}_+^I$. Note that $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote the sets of nonnegative and positive numbers, respectively. $\log(\cdot)$ is the element-wise logarithmic function, $\boldsymbol{x}\in\mathbb{R}_{++}^I$ is a positive vector (i.e., $x_i>0\ \forall i$), and its PDF is given by Eq. (1): $p(\boldsymbol{x})=\exp(-E_\Gamma(\boldsymbol{x}))/Z$.

The seemingly strange constraints on the trainable parameters ($U_{ij}<0, u_i\le 0, S_{ij}>0, s_i\ge -1\ \forall i,j$) come from the corresponding distribution as follows. For calculating the conditional distribution of x_i given the other units, let Eq. (12) be rewritten by gathering the variables as

$$E_\Gamma(x_i; x_j\ \forall j\neq i) = \beta_i x_i - (\alpha_i-1)\log x_i - r_i, \tag{13}$$

where $E_\Gamma(x_i; x_j\ \forall j\neq i)$ represents the energy function for x_i obtained by fixing the other variables x_j ($\forall j\neq i$), α_i and β_i are

the terms related to $\log(x_i)$ and x_i, respectively,

$$\alpha_i = 1 + s_i + \sum_{j \neq i} S_{ij} \log x_j, \quad \beta_i = -u_i - \sum_{j \neq i} U_{ij} x_j, \tag{14}$$

and r_i is the term unrelated to $\log(x_i)$ and x_i.[3] Then, the conditional distribution $p(x_i|x_j \, \forall j \neq i)$ can be calculated as

$$p(x_i|x_j \, \forall j \neq i) = \mathcal{G}(x_i; \alpha_i, \beta_i), \tag{16}$$

where $\mathcal{G}(x; \alpha, \beta)$ is the gamma distribution,

$$\mathcal{G}(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \tag{17}$$

$\Gamma(\,\cdot\,)$ is the gamma function, and the computation is detailed in the footnote.[4] This is the reason why we termed the proposed Boltzmann machine as *gamma* Boltzmann machine. By definition of the gamma distribution, the parameters are constrained to be positive: $\alpha_i > 0$ and $\beta_i > 0$. Therefore, from Eq. (14), the parameters of the proposed Boltzmann machine are constrained as $U_{ij} < 0$, $u_i \leq 0$, $S_{ij} > 0$, and $s_i \geq -1 \; \forall i, j$. Owing to the gamma distribution, this model naturally forces the variables $\boldsymbol{x}$ to be positive. By introducing the log-related parameters $\mathbf{S}$ and $\boldsymbol{s}$ in addition to the ordinary Boltzmann machine in Eq. (2), the proposed model can learn a PDF with consideration of the logarithmic scale.

B. Transition From Gamma Boltzmann Machine to RBM

By separating $\boldsymbol{x}$ into visible and hidden units and imposing restriction, we can obtain an RBM based on the above gamma Boltzmann machine. Yet, some additional care is necessary.

Because of the logarithmic function in Eq. (12), all units must be positive. In our model, data are assumed to be positive, $\boldsymbol{v} \in \mathbb{R}^D_{++}$, and the hidden variables are binary, $\boldsymbol{h} \in \{0, 1\}^H$. However, this assumption cannot be directly accepted because $\log(\boldsymbol{h})$ takes $-\infty$ whenever $\boldsymbol{h}$ contains 0. Therefore, we consider the (element-wise) exponential function that makes the values positive: $\exp(\boldsymbol{h}) \in \{1, \mathrm{e}\}^H$.

With this modification, an energy function is defined as

$$\begin{aligned} E_{\Gamma\mathrm{Bpre}}(\boldsymbol{v}, \boldsymbol{h}) = & -\boldsymbol{v}^\mathsf{T} \mathbf{W} \exp(\boldsymbol{h}) - \boldsymbol{b}^T \boldsymbol{v} - \boldsymbol{c}^\mathsf{T} \exp(\boldsymbol{h}) \\ & - \log(\boldsymbol{v})^\mathsf{T} (\mathbf{V}\boldsymbol{h} - \mathbf{1}) - \boldsymbol{d}^\mathsf{T} \boldsymbol{h}, \end{aligned} \tag{21}$$

[3]The explicit form of the term unrelated to $\log(x_i)$ and x_i is given by

$$r_i = \sum_{j \neq i} (u_j x_j + s_j \log x_j) + \frac{1}{2} \sum_{\neq i} (U_{i'j'} x_{i'} x_{j'} + S_{i'j'} \log x_{i'} \log x_{j'}), \tag{15}$$

where the second summation is taken for $i,' j'$ that are not equals to i.

[4]Since $p(x_i|x_j \, \forall j \neq i) = \frac{1}{Z} \mathrm{e}^{-E_\Gamma(x_i; x_j \, \forall j \neq i)}$ by definition, we obtain

$$\frac{1}{Z} \mathrm{e}^{-E_\Gamma(x_i; x_j \, \forall j \neq i)} = \frac{\mathrm{e}^{-\beta_i x_i + (\alpha_i - 1) \log x_i + r_i}}{\int_0^\infty \mathrm{e}^{-\beta_i x_i + (\alpha_i - 1) \log x_i + r_i} \, \mathrm{d}x_i}, \tag{18}$$

$$= \frac{x_i^{\alpha_i - 1} \mathrm{e}^{-\beta_i x_i}}{\int_0^\infty x_i^{\alpha_i - 1} \mathrm{e}^{-\beta_i x_i} \, \mathrm{d}x_i}, \tag{19}$$

$$= \frac{\beta_i^{\alpha_i}}{\Gamma(\alpha_i)} x_i^{\alpha_i - 1} \mathrm{e}^{-\beta_i x_i}, \tag{20}$$

which is the gamma distribution, and hence $p(x_i|x_j \, \forall j \neq i) = \mathcal{G}(x_i; \alpha_i, \beta_i)$.

that can be derived from Eq. (12) by inserting

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{v} \\ \exp(\boldsymbol{h}) \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{O} & \mathbf{W} \\ \mathbf{W}^\mathsf{T} & \mathbf{O} \end{bmatrix}, \qquad \boldsymbol{u} = \begin{bmatrix} \boldsymbol{b} \\ \boldsymbol{c} \end{bmatrix}, \tag{22}$$

$$\mathbf{S} = \begin{bmatrix} \mathbf{O} & \mathbf{V} \\ \mathbf{V}^\mathsf{T} & \mathbf{O} \end{bmatrix}, \qquad \boldsymbol{s} = \begin{bmatrix} -\mathbf{1} \\ \boldsymbol{d} \end{bmatrix}, \tag{23}$$

where $-\mathbf{W} \in \mathbb{R}^{D \times H}_{++}$, $\mathbf{V} \in \mathbb{R}^{D \times H}_{++}$, $\mathbf{1} \in \{1\}^D$, $-\boldsymbol{b} \in \mathbb{R}^D_+$, $-\boldsymbol{c} \in \mathbb{R}^H_+$ $\boldsymbol{d} \in \mathbb{R}^H$, and the joint density function of $\boldsymbol{v}$ and $\boldsymbol{h}$ is given as in Eq. (3): $p(\boldsymbol{v}, \boldsymbol{h}) = \exp(-E_{\Gamma\mathrm{Bpre}}(\boldsymbol{v}, \boldsymbol{h}))/Z$. Note that the upper part of $\boldsymbol{s}$ is fixed to $-\mathbf{1}$ because of the condition required by the gamma distribution.

C. Proposed Gamma-Bernoulli RBM

Based on the above energy function, we propose gamma-Bernoulli RBM through the following simplification.[5]

One drawback of the above RBM is that $\exp(\boldsymbol{h})$ cannot take zero, i.e., all column vectors of $\mathbf{W}$ must be active regardless of the hidden state. To circumvent this unfavorable situation, we consider a specific choice for the bias vector: $\boldsymbol{b} = -\mathbf{W}\mathbf{1}$ (multiplication of the matrix $\mathbf{W}$ and the all-one vector $\mathbf{1}$). This choice of $\boldsymbol{b}$ simplifies the above energy function as

$$\begin{aligned} E_{\Gamma\mathrm{B}}(\boldsymbol{v}, \boldsymbol{h}) = & -\boldsymbol{v}^\mathsf{T} \mathbf{W} (\exp(\boldsymbol{h}) - \mathbf{1}) - \boldsymbol{c}^\mathsf{T} \exp(\boldsymbol{h}) \\ & - \log(\boldsymbol{v})^\mathsf{T} (\mathbf{V}\boldsymbol{h} - \mathbf{1}) - \boldsymbol{d}^\mathsf{T} \boldsymbol{h}, \end{aligned} \tag{24}$$

which is more symmetric for linear and logarithmic domains. By this simplification, the elements of $\exp(\boldsymbol{h}) - \mathbf{1}$ can take zero, and hence the hidden variables can work as binary selectors like in the usual RBMs. Note that we can further simplify this energy function by omitting $\boldsymbol{c}$ as shown in the experimental section (see Table 1).

The proposed RBM based on Eq. (24) is termed *gamma-Bernoulli RBM* since its conditional distributions are given by

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{G}(\boldsymbol{v}; \mathbf{V}\boldsymbol{h}, -\mathbf{W}(\exp(\boldsymbol{h}) - \mathbf{1})), \tag{25}$$

$$\begin{aligned} p(\boldsymbol{h}|\boldsymbol{v}) = & \mathcal{B}(\boldsymbol{h}; f_\sigma[\,(\mathrm{e} - 1)(\boldsymbol{c} + \mathbf{W}^\mathsf{T} \boldsymbol{v}) \\ & + \boldsymbol{d} + \mathbf{V}^\mathsf{T} \log(\boldsymbol{v})\,]), \end{aligned} \tag{26}$$

where $\mathcal{G}(\,\cdot\,; \boldsymbol{\alpha}, \boldsymbol{\beta})$ for a vector input represents the element-wise i.i.d. gamma distribution with a shape-parameter vector $\boldsymbol{\alpha} \in \mathbb{R}^D_{++}$ and a rate-parameter vector $\boldsymbol{\beta} \in \mathbb{R}^D_{++}$, i.e., $\mathcal{G}(\boldsymbol{x}; \boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_i \mathcal{G}(x_i; \alpha_i, \beta_i)$. This relation shows that the proposed gamma-Bernoulli RBM handles data through the gamma distribution. Note that the Bernoulli distribution in Eq. (26) arises from the binary assumption of $\boldsymbol{h}$. By removing the binary assumption, we can obtain a gamma-gamma RBM that handles both visible and hidden units by the gamma distribution (see Appendix B).

The gamma distribution is a natural choice for modeling positive data. Furthermore, some research has reported that the gamma distribution can approximate the distribution of speech signals better than the Gaussian distribution regardless of the

[5]Note that a variant of gamma-Bernoulli RBM has been proposed for synthetic aperture radar image classification [33]. Its focus is not on handling data in logarithmic domain, and therefore it is essentially different from the proposed RBM that equally treats linear and logarithmic domains.

type of speech parameterization [34]–[37]. Thus, the proposed gamma-Bernoulli RBM should be more suitable for modeling magnitude spectra than the Gaussian-Bernoulli RBM.

D. Implementation of Gamma-Bernoulli RBM

Since the gamma-Bernoulli RBM has specific conditions on trainable parameters, they must be handled with care. We ensure the conditions by the following techniques.

From Eq. (25), $\mathbf{V}\boldsymbol{h}$ and $-\mathbf{W}(\exp(\boldsymbol{h})-\mathbf{1})$ correspond to the parameters of the gamma distribution ($\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, respectively) that are positive by definition. Hence, the elements of $\mathbf{V}$ and $-\mathbf{W}$ must be positive for satisfying the definition of the gamma distribution because both $\boldsymbol{h}$ and $(\exp(\boldsymbol{h})-\mathbf{1})$ are nonnegative. To ensure positivity of $\mathbf{V}$ and $-\mathbf{W}$ without causing instability of training, we parameterize them as follows [18]:

$$\mathbf{W} = -f_+(\widetilde{\mathbf{W}}), \qquad \mathbf{V} = f_+(\widetilde{\mathbf{V}}), \tag{27}$$

where $\widetilde{\mathbf{W}} \in \mathbb{R}^{D\times H}$, $\widetilde{\mathbf{V}} \in \mathbb{R}^{D\times H}$, and $f_+(\,\cdot\,)$ is a positive function. In this paper, we consider the exponential and softplus functions as examples of f_+, respectively given as

$$f_+(\boldsymbol{x}) = \exp(\boldsymbol{x}), \qquad f_+(\boldsymbol{x}) = \log(\mathbf{1}+\exp(\boldsymbol{x})). \tag{28}$$

The choice of these positive functions will be investigated in the experimental section.

Even though positivity of $\mathbf{V}$ and $-\mathbf{W}$ is ensured by the above parametrization, $\mathbf{V}\boldsymbol{h}$ and $-\mathbf{W}(\exp(\boldsymbol{h})-\mathbf{1})$ become zero whenever $\boldsymbol{h}=\mathbf{0}$. In order to avoid such situation, the vector $\boldsymbol{s}$ given in Eq. (23) may be modified as $\boldsymbol{s} = [\,(-\mathbf{1}+\varepsilon)^{\mathsf{T}}, \boldsymbol{d}^{\mathsf{T}}\,]^{\mathsf{T}}$ with a small positive constant $\varepsilon > 0$. This addition makes the shape parameter of the gamma distribution in Eq. (25), $\boldsymbol{\alpha} = \mathbf{V}\boldsymbol{h}+\varepsilon$, always positive as required by the definition. In the same way, the rate parameter can be forced positive by modifying it to $\boldsymbol{\beta} = -\mathbf{W}(\exp(\boldsymbol{h})-\mathbf{1}+\varepsilon)$. However, based on our preliminary study, these modifications have no impact in practice because $\boldsymbol{h}=\mathbf{0}$ rarely happens. Hence, we did not add ε in experiments to reduce the number of tunable parameters.

E. Objective Function and Parameter Optimization

As the conventional Boltzmann machines, the objective of the proposed RBM is to maximize the log-likelihood:

$$L(\{\boldsymbol{v}^{(n)}\}_n) = \frac{1}{N}\sum_n \log(p(\boldsymbol{v}^{(n)})) \tag{29}$$

$$= \frac{1}{N}\sum_n \log\Big(\sum_{\boldsymbol{h}^{(n)}} p(\boldsymbol{v}^{(n)}, \boldsymbol{h}^{(n)})\Big) \tag{30}$$

$$= \frac{1}{N}\sum_n \log\Big(\sum_{\boldsymbol{h}^{(n)}} \mathrm{e}^{-E(\boldsymbol{v}^{(n)}, \boldsymbol{h}^{(n)})}\Big) - \log Z, \tag{31}$$

where $\boldsymbol{v}^{(n)}$ and $\boldsymbol{h}^{(n)}$ are the nth training data and the corresponding hidden variables, respectively, and $\sum_{\boldsymbol{h}^{(n)}}$ represents marginalization over all possible states of $\boldsymbol{h}^{(n)}$.

For optimizing an RBM, the gradient of the log-likelihood function w.r.t. the parameters $\boldsymbol{\theta} = (\widetilde{\mathbf{W}}, \widetilde{\mathbf{V}}, \boldsymbol{c}, \boldsymbol{d})$ is required. Although it can be explicitly written as

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \Big\langle -\frac{\partial E}{\partial \boldsymbol{\theta}} \Big\rangle_{\mathrm{data}} - \Big\langle -\frac{\partial E}{\partial \boldsymbol{\theta}} \Big\rangle_{\mathrm{model}}, \tag{32}$$

this gradient is practically intractable owing to the second term, where $\langle\,\cdot\,\rangle_{\mathrm{data}}$ and $\langle\,\cdot\,\rangle_{\mathrm{model}}$ represent the expectations on data and model distributions, respectively. Therefore, as usual in the conventional Boltzmann machines, the contrastive divergence method [38] is applied to approximate the gradient:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} \approx \Big\langle -\frac{\partial E}{\partial \boldsymbol{\theta}} \Big\rangle_{\mathrm{data}} - \Big\langle -\frac{\partial E}{\partial \boldsymbol{\theta}} \Big\rangle_{\mathrm{recon}}, \tag{33}$$

where $\langle\,\cdot\,\rangle_{\mathrm{recon}}$ is the expectation on the reconstructed data usually obtained through the Gibbs sampling.

The negative partial gradients of the energy function in Eq. (24) w.r.t. the vectors $\boldsymbol{c}$ and $\boldsymbol{d}$ are obtained as follows:

$$-\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \boldsymbol{c}} = \exp(\boldsymbol{h}), \qquad -\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \boldsymbol{d}} = \boldsymbol{h}. \tag{34}$$

The gradients w.r.t. $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{V}}$ depend on their parametrization in Eq. (27). If $f_+(\,\cdot\,) = \exp(\,\cdot\,)$, their gradients are given by

$$-\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \widetilde{\mathbf{W}}} = \exp(\widetilde{\mathbf{W}}) \circ (\boldsymbol{v}(\mathbf{1}-\exp(\boldsymbol{h}))^{\mathsf{T}}), \tag{35}$$

$$-\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \widetilde{\mathbf{V}}} = \exp(\widetilde{\mathbf{V}}) \circ (\log(\boldsymbol{v})\,\boldsymbol{h}^{\mathsf{T}}), \tag{36}$$

where $\circ$ denotes the element-wise multiplication. Similarly, if $f_+(\,\cdot\,) = \log(\mathbf{1}+\exp(\,\cdot\,))$, their gradients are given by

$$-\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \widetilde{\mathbf{W}}} = f_\sigma[\widetilde{\mathbf{W}}] \circ (\boldsymbol{v}(\mathbf{1}-\exp(\boldsymbol{h}))^{\mathsf{T}}), \tag{37}$$

$$-\frac{\partial E_{\Gamma\mathrm{B}}}{\partial \widetilde{\mathbf{V}}} = f_\sigma[\widetilde{\mathbf{V}}] \circ (\log(\boldsymbol{v})\,\boldsymbol{h}^{\mathsf{T}}). \tag{38}$$

By using these formulae, the gamma-Bernoulli RBM can be trained with a gradient-based optimization algorithm. Note that, when $f_+(\,\cdot\,) = \exp(\,\cdot\,)$, gradients in Eqs. (35) and (36) can be computed using $-\mathbf{W}$ and $\mathbf{V}$, instead of calculating $\exp(\widetilde{\mathbf{W}})$ and $\exp(\widetilde{\mathbf{V}})$, that can reduce the computational cost.

F. Some Extensions of the Proposed Boltzmann Machines

In the next section, we will compare the proposed gamma-Bernoulli RBM with its variants. Since these variants are proposed merely for experimental investigation, we leave their details in Appendices and just briefly mention them here.

Firstly, a gamma-Bernoulli RBM that can automatically balance the contribution from linear and logarithmic scales is proposed in Appendix A. It can be expected that the preference of the scales might depend on a task: some task prefers a linear scale more than a logarithmic scale, and vice versa. A gamma-Bernoulli RBM in Appendix A introduces a trainable trade-off parameter that balances the importance of linear and logarithmic scales. The effect of this extension will be experimentally investigated in Section IV-F.

Secondly, a gamma-gamma RBM that handles both visible and hidden units by the gamma distribution is proposed in Appendix B. Unlike the gamma-Bernoulli RBM, the gamma-gamma RBM considers real-valued hidden units, which have

a better expressivity than binary units. Its performance will be experimentally investigated in Section IV-G.

IV. Experiments

In this section, the properties of the proposed gamma-Bernoulli RBM are investigated, and then its performance is compared with those of the Gaussian-Bernoulli RBM and VAE by speech representation experiments.

Firstly, the experimental conditions are summarized in Section IV-A. Then, the properties of the proposed RBM are investigated in Section IV-B. Using the best setup revealed in Section IV-B, the proposed RBM is compared with the conventional RBM in Section IV-C. It is also compared with VAEs in Section IV-D. Moreover, discussion on data compression and binarization are provided in Section IV-E. Finally, some extensions of the proposed RBM are tested in Sections IV-F and IV-G for further discussion.

A. Experimental Configuration

In the experiments, the ATR speech corpus (set B, speaker FTK) was utilized. The speech signals of 50 sentences (SDA) were utilized for training, while the other 53 sentences (SDJ) were used for evaluation. Those signals originally sampled at 20 kHz were downsampled to 16 kHz for speeding up the computation. The short-time Fourier transform (STFT) was implemented with a 256-sample-long Blackman window and a hop size of 64 samples. The 129-dimensional data vector $\boldsymbol{v}^{(n)}$ was calculated by taking the absolute value of the spectrum of each windowed segment. After discarding silent segments, the number of the data samples for training was 51 197.

Utilized data were normalized so that the data distribution was standardized. That is, as usual, each dimension was normalized so that the data were distributed with center 0 and standard deviation 1. Evaluation was performed after canceling the effect of normalization by the inverse operation.

All RBMs were trained by the Adam optimizer [39] with a batch size 100 and a learning rate 0.01. After training with 100 epochs, magnitude spectra of a signal in the evaluation dataset were encoded and reconstructed using the trained RBMs by calculating expectation of $\boldsymbol{v}$ from expectation of the encoded code $\boldsymbol{h}$ of the inputted data samples, i.e., reconstruction is obtained from $p(\boldsymbol{v}|\mathbb{E}_{p(\boldsymbol{h}|\boldsymbol{v}^{(n)})}[\boldsymbol{h}])$. The reconstructed magnitude spectra were evaluated by a subjective test or objective measures: MSE (mean-squared error), PESQ (perceptual evaluation of speech quality) and STOI (short-time objective intelligibility), where time-domain signals were calculated using the inverse STFT with the original phase.

B. Properties of the Proposed Gamma-Bernoulli RBM

At first, we investigated the proposed gamma-Bernoulli RBM from three viewpoints: (1) combination of trainable parameters $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{d}$; (2) choice of f_+ for parametrizing $\mathbf{W}$ and $\mathbf{V}$; and (3) normalization of data.

1) Combination of the Bias Parameters $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{d}$: The proposed RBM in Eq. (24) has three bias parameters $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{d}$ [recall that $\boldsymbol{b}$ is explicitly defined in Eq. (21) and fixed to $\boldsymbol{b} = -\mathbf{W}\mathbf{1}$ in Eq. (24)]. Note that our preliminary version [29] did not contain $\boldsymbol{b}$, i.e., we decided to make the model more general. Since the performance should depend on the choice of these parameters, their effect is investigated here.

TABLE I
PESQ of the Proposed RBMs (800 Hidden Units) Using Different Combinations of Bias Parameters ($\boldsymbol{c}$, $\boldsymbol{d}$, and $\boldsymbol{b} = -\mathbf{W}\mathbf{1}$) in Eq. (24)

Use $\boldsymbol{b}$	Use $\boldsymbol{c}$	Use $\boldsymbol{d}$	PESQ
	✓	✓	3.31 [29]
✓	✓	✓	4.14
✓		✓	**4.23**
✓	✓		4.20
✓			2.80

TABLE II
PESQ of the Proposed RBMs (800 Hidden Units) Using Different Combinations of Positive Functions f_+ in Eq. (28)

f_+ for $\mathbf{W}$	f_+ for $\mathbf{V}$	PESQ	$\|\mathbf{W}\|_F$	$\|\mathbf{V}\|_F$
exp	exp	4.21	938.99	119.00
exp	softplus	4.18	770.49	104.41
softplus	exp	**4.23**	613.63	120.72
softplus	softplus	**4.23**	613.29	112.34

Table 1 shows PESQ scores (averaged over all evaluation data) for 5 variants of the proposed gamma-Bernoulli RBM. The first row indicates our preliminary version in [29], and the other 4 rows represents RBMs newly developed in this paper. From the result, it can be seen that $\boldsymbol{b}$ $(= -\mathbf{W}\mathbf{1})$ is essential for achieving a better performance, but using only $\boldsymbol{b}$ cannot learn data representation properly. One of $\boldsymbol{c}$ or $\boldsymbol{d}$ should be contained in the gamma-Bernoulli RBM in addition to $\boldsymbol{b}$, but all of them does not have to be used. Since discarding a trainable parameter can reduce the computational complexity, we utilize $\boldsymbol{b}$ and $\boldsymbol{d}$ and omit $\boldsymbol{c}$ hereafter.

2) Choice of the Positive Function $f_+(\cdot)$: As in Eq. (28), we consider two choices for f_+ in this paper: the exponential and softplus functions. To see the effect of these positive functions, we compared all combinations of them (since they are applied to $\mathbf{W}$ and $\mathbf{V}$, there are 4 combinations).

Table 2 shows PESQ scores for all combinations of f_+ and $\mathbf{W}, \mathbf{V}$. From the result, it can be seen that the choice of f_+ has little impact on the reconstruction performance in terms of PESQ. However, it had some impact on the Frobenius norm of $\mathbf{W}$ and $\mathbf{V}$ as shown on the right side of Table 2. For stable training, the norm of parameters should not be excessively large. Since $\mathbf{W}$ with the exponential function resulted in a larger norm value, the softplus function seems a better choice for $\mathbf{W}$. Hereafter, we utilize the softplus function for both $\mathbf{W}$ and $\mathbf{V}$ (note that the previous subsection also used it).

3) Normalization of Data: Data normalization is the standard strategy to make training easier. As mentioned in Section IV-A, experiments in this paper utilized the usual standardization (mean 0 and standard deviation 1). Here, we compared normalization methods to see whether this choice is right or not. For the proposed gamma-Bernoulli RBM, we additionally

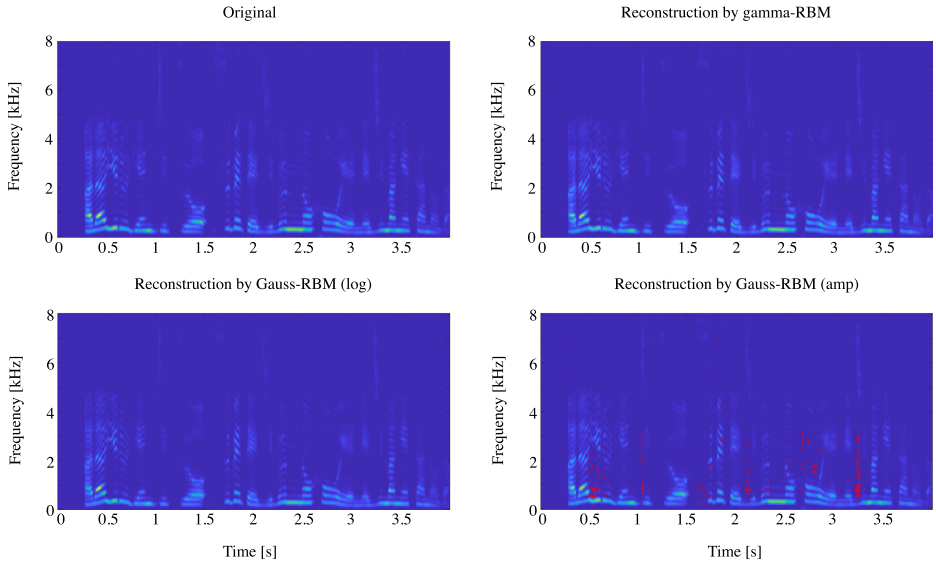

Fig. 2. Original and reconstructed (linear-)magnitude spectra. The reconstructed spectra were obtained by the proposed (`gamma-RBM (H800)`) and by the conventional (`Gauss-RBM (log, H800)` and `Gauss-RBM (amp, H800)`) models. The red points represent negative values which should not exist.

TABLE III
PERFORMANCE OF RBMS (800 HIDDEN UNITS) USING DIFFERENT NORMALIZATION METHODS [MAX1 REPRESENTS EQ. (39)]

Method	Normalization	PESQ	$\|\mathbf{W}\|_F$	$\|\mathbf{V}\|_F$
gamma-RBM	w/o normalization	4.19	2800.93	100.35
	standardization	4.23	613.29	112.34
	max1	4.20	4988.85	78.29
Gauss-RBM (log)	w/o normalization	2.89	117.93	N/A
	standardization	4.10	239.85	N/A
Gauss-RBM (amp)	w/o normalization	3.27	612.94	N/A
	standardization	3.15	401.76	N/A

considered the following normalization:

$$\widetilde{x} = \hat{\beta} x = \alpha x / \overline{x}, \tag{39}$$

where $\overline{x}$ denotes mean of x, and $\hat{\beta}$ is the maximum-likelihood estimation of β. This is because the gamma distribution, assumed as $\mathcal{G}(x; \alpha, \hat{\beta})$, becomes the standard form, $\mathcal{G}(\widetilde{x}; \alpha, 1)$. In this experiment, $\alpha = 1$ was used for the normalization.

Table 3 shows PESQ scores for different normalization methods. This table shows not only the proposed RBM (`gamma-RBM`) but also the Gaussian-Bernoulli RBM (`Gauss-RBM`) for comparison. Furthermore, two variations for the Gaussian-Bernoulli RBM were considered: `(log)` represents that all data (magnitude spectra) were computed with the logarithmic function, while `(amp)` denotes those without logarithm (raw magnitude spectra). From the result, it can be seen that the performance of the Gaussian-Bernoulli RBM heavily depends on the presence of the normalization. In contrast, the proposed gamma-Bernoulli RBM was able to perform well regardless of the normalization. This might be because $\mathbf{W}$ and $\mathbf{V}$ can balance the scale of data, which is indicated by the Frobenius norm on the right side of Table 3.

C. Performance Comparison With the Conventional RBM

Next, the proposed gamma-Bernoulli RBM is compared with the conventional Gaussian-Bernoulli RBM. Here, comparison is made from four aspects: (1) qualitative comparison of magnitude spectra; (2) quantitative comparison using PESQ, STOI and MSE; (3) comparison of MSE during the training; and (4) subjective comparison of reconstructed signals.

In the following figures and table, the proposed gamma-Bernoulli RBM is denoted by `gamma-RBM`, while the conventional Gaussian-Bernoulli RBM is denoted by `Gauss-RBM`. For `Gauss-RBM`, `log` and `amp` respectively represent the RBM trained with and without the logarithmic transformation of data. In other words, `Gauss-RBM (amp)` was trained and utilized with raw magnitude spectra, while `Gauss-RBM (log)` was trained and utilized in the logarithmic domain (with log-magnitude spectra). The number of hidden units is represented with `H` as the indicator, e.g., `H800` represents that the number of hidden unit was 800.

1) Qualitative Comparison of Reconstructed Spectrograms: Magnitude spectra reconstructed by the RBMs are shown in Figs. 2 and 3 using the linearly and logarithmically scaled colors, respectively. In Fig. 2, negative values, which never exist in magnitude spectra, are marked by red points. From Fig. 2, it can be seen that `Gauss-RBM (amp)` produced negative values. In contrast, both `Gauss-RBM (log)` and `gamma-RBM` did not produce any negative value. This result indicates that the proposed RBM can properly handle positive data without producing a negative value.

From Fig. 3, it is obvious that `Gauss-RBM (amp)` was not able to reconstruct small values (illustrated by light green). Comparing `gamma-RBM` with `Gauss-RBM (log)`, `gamma-RBM` produced smoother spectra for very small values (illustrated by dark blue). This slight difference should be evaluated using quantitative metrics.

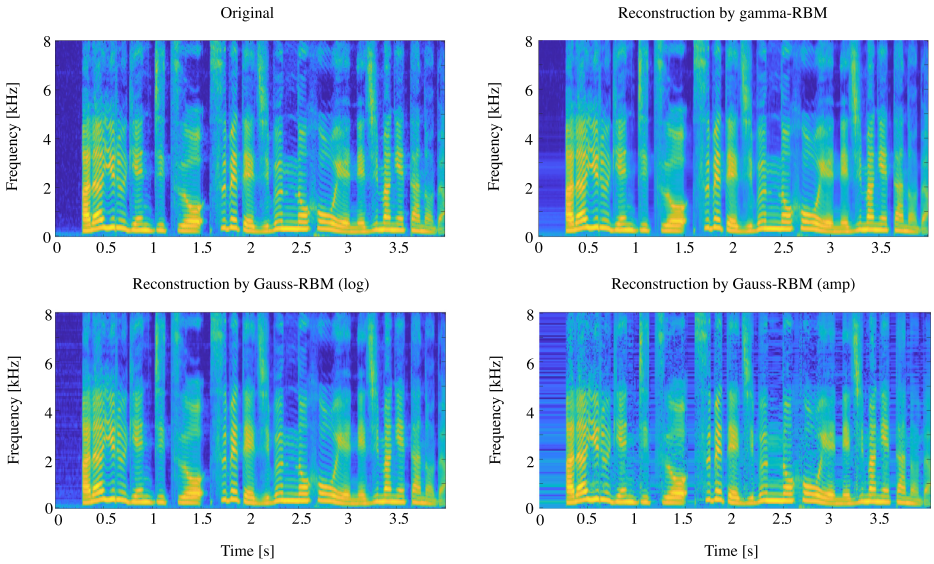

Fig. 3. Original and reconstructed log-magnitude spectra. The reconstructed spectra were obtained by the proposed (`gamma-RBM (H800)`) and by the conventional (`Gauss-RBM (log, H800)` and `Gauss-RBM (amp, H800)`) models. Color range is 84.0 dB (from dark blue to bright yellow).

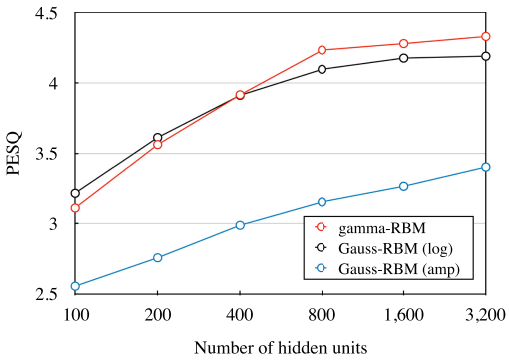

Fig. 4. PESQ scores for the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black, and `Gauss-RBM (amp)`, blue) models defined with various numbers of hidden units.

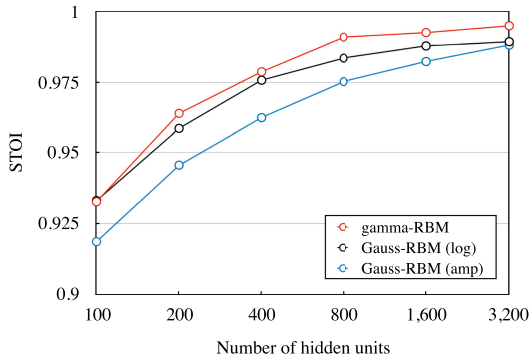

Fig. 5. STOI scores for the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black, and `Gauss-RBM (amp)`, blue) models defined with various numbers of hidden units.

2) Quantitative Comparison of Reconstructed Signals: The proposed `gamma-RBM` was compared with the conventional `Gauss-RBM`s using PESQ, STOI and two MSEs defined by

$$\mathrm{MSE}_{\mathrm{amp}} = \frac{1}{N}\sum_{n=1}^{N} \left\| \boldsymbol{v}^{(n)} - \hat{\boldsymbol{v}}^{(n)} \right\|_2^2, \tag{40}$$

$$\mathrm{MSE}_{\mathrm{log}} = \frac{1}{N}\sum_{n=1}^{N} \left\| \log|\boldsymbol{v}^{(n)}| - \log|\hat{\boldsymbol{v}}^{(n)}| \right\|_2^2, \tag{41}$$

where $\|\cdot\|_2$ is the Euclidean norm, $\boldsymbol{v}^{(n)}$ is a magnitude spectrum in the evaluation dataset, and $\hat{\boldsymbol{v}}^{(n)}$ is its reconstruction by an RBM, which may have negative values (see Fig. 2).

PESQ and STOI are illustrated in Figs. 4 and 5, respectively. From Fig. 4, `Gauss-RBM (amp)` (drawn by the blue line) performed worst for all numbers of hidden units. Compared PESQ of `gamma-RBM` (red line) with that of `Gauss-RBM (log)` (black line), `Gauss-RBM (log)` marginally outperformed `gamma-RBM` when the number of hidden units was small (`H100` and `H200`). The proposed `gamma-RBM` outperformed the conventional `Gauss-RBM (log)` when the number of hidden units was large enough (`H800`, `H1600` and `H3200`). Note that Fig. 3 show the case `H800`. It indicates that the slight difference in very small values in Fig. 3 was not so important in terms of PESQ. `Gauss-RBM (log)` models very small values with equal importance compared to larger values, and hence it suffers from $v_i^{(n)} \approx 0$ which becomes outlier in the logarithmic domain.

To see the performance of RBMs trained and tested with more speakers, the RBMs were also compared by using the `DR1` subset of the TIMIT dataset. The training data consisted of 380 utterances from 38 speakers, and the test data were 110 utterances from the other 11 speakers. The other training conditions were the same. Table 4 shows the averaged scores of PESQ and STOI for the RBMs (`H800`). As can be seen from the table, the proposed RBM was able to outperform the conventional RBM for this case as well.

$\mathrm{MSE}_{\mathrm{amp}}$ and $\mathrm{MSE}_{\mathrm{log}}$ are illustrated in Figs. 6 and 7, respectively. As shown in the figures, `Gauss-RBM (amp)` reduced $\mathrm{MSE}_{\mathrm{amp}}$ successfully, while `Gauss-RBM (log)` reduced

TABLE IV
PESQ AND STOI OF THE RBMS (800 HIDDEN UNITS) TRAINED AND TESTED BY USING THE TIMIT DATASET CONSISTED OF MANY SPEAKERS

Methods	PESQ	STOI
gamma-RBM	**4.17**	**1.00**
Gauss-RBM (log)	4.07	0.99
Gauss-RBM (amp)	2.69	0.92

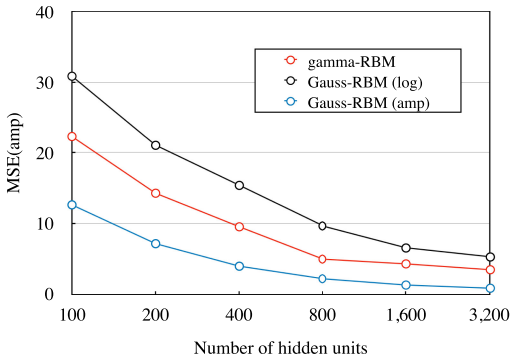

Fig. 6. MSE of magnitude spectra obtained by the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black, and `Gauss-RBM (amp)`, blue) models defined with various numbers of hidden units.

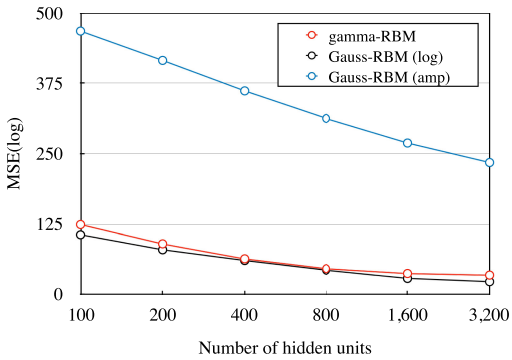

Fig. 7. MSE of magnitude spectra obtained by the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black and `Gauss-RBM (amp)`, blue) models defined with various numbers of hidden units.

$\mathrm{MSE}_{\log}$. This is a natural consequence of applying `Gauss-RBM` in the linear or logarithmic domain. The proposed `gamma-RBM` was able to reduce $\mathrm{MSE}_{\log}$ like `Gauss-RBM (log)` as in Fig. 7. Furthermore, `gamma-RBM` was also able to reduce $\mathrm{MSE}_{\mathrm{amp}}$ to some extent (Fig. 6).

3) MSEs During Training: $\mathrm{MSE}_{\mathrm{amp}}$ and $\mathrm{MSE}_{\log}$ during the training are illustrated in Figs. 8 and 9, respectively. From the figures, `Gauss-RBM`s (black and blue) was able to rapidly decrease one of the MSEs. In other words, `Gauss-RBM (amp)` rapidly decreased $\mathrm{MSE}_{\mathrm{amp}}$ as in Fig. 8, while `Gauss-RBM (log)` rapidly decreased $\mathrm{MSE}_{\log}$ as in Fig. 9. However, they were slow to decrease the other MSE that was not considered by their definition. In contrast, the proposed `gamma-RBM` (red) was able to rapidly decrease both MSEs within, say, 10 epochs as in Figs. 8 and 9. These results indicate that the proposed `gamma-RBM` can model data with consideration of both linear and logarithmic scales.

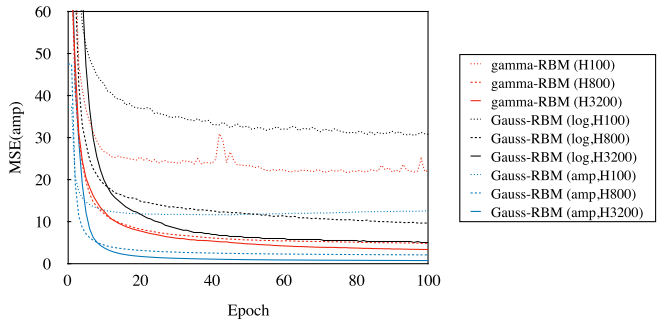

Fig. 8. MSE of magnitude spectra during training of the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM`, black and blue) models. The number after `H` indicate the number of hidden units.

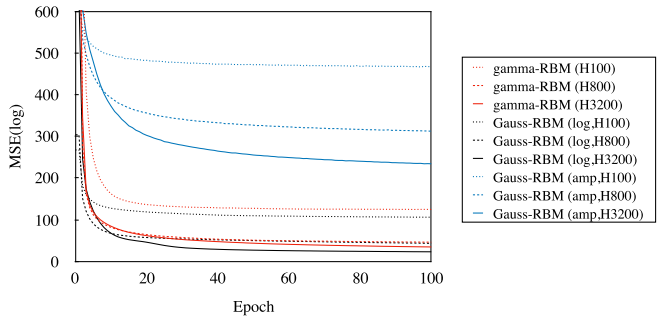

Fig. 9. MSE of magnitude spectra during training of the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM`, black and blue) models.

TABLE V
RESULT OF THE SUBJECTIVE TEST WITH 43 PARTICIPANTS (MOS OF THE RECONSTRUCTED SIGNALS). CI REPRESENTS THE CONFIDENCE INTERVAL

Method	MOS ($\pm$95%CI)
gamma-RBM (H800)	**4.332** $\pm$ 0.091
Gauss-RBM (log, H800)	4.199 $\pm$ 0.095
WORLD [40]	3.043 $\pm$ 0.117
Original	4.432 $\pm$ 0.085

4) Subjective Evaluation: A subjective test was performed to compare the sound quality of reconstructed signals. We asked 43 participants to evaluate the original and reconstructed speech signals using the following five labels: Excellent (5), Good (4), Fair (3), Poor (2), and Bad (1). After the evaluation, mean opinion scores (MOS) was calculated. As the reference, the original signals and those analyzed and synthesized by WORLD [40] were also evaluated.[6] The number of hidden units of RBMs was set to 800 because, from the viewpoint of information compression, using excessive number of hidden units is not of interest.

The evaluated scores are shown in Table 5. The 95% confidence interval is also shown on the side of each score. From the table, it can be seen that the proposed `gamma-RBM` significantly outperformed the conventional `Gauss-RBM (log)`.

[6]Note that the comparison between RBMs and WORLD is not fair. The reconstructed signals by RBMs utilized phase of the original signals, while WORLD did not. Therefore, the score of WORLD is shown merely as a reference. We focused on modeling of magnitude spectra in this paper, and modeling of phase is left as a future work.

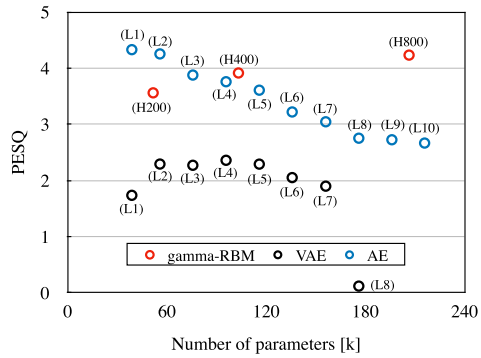

Fig. 10. PESQ scores for the proposed RBMs (red) and the AEs/VAEs (blue and black, respectively) w.r.t. the number of parameters.

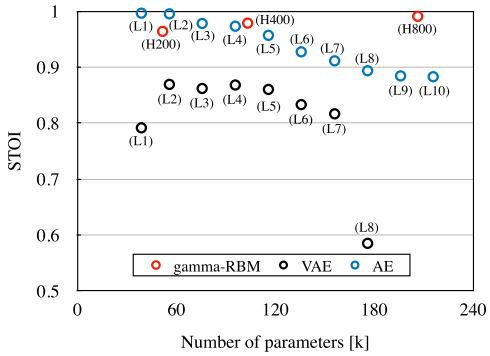

Fig. 11. STOI scores for the proposed RBMs (red) and the AEs/VAEs (blue and black, respectively) w.r.t. the number of parameters.

Moreover, the sound quality of `gamma-RBM` was comparable to the original signals. This result indicates that the proposed RBM can effectively model audio signals.

D. Performance Comparison With Deep Neural Networks

Here, the proposed RBM is compared with VAEs in terms of the objective measures and the number of parameters. In this experiment, the input 129-dimensional vector was handled by the fully-connected layers whose output dimension was 100. By stacking the fully-connected layers with ReLU as the activation function, the networks were made deeper. The networks are distinguished by their depth, which is indicated by the number after the indicator `L`, e.g., `L3` consisted of 3 fully-connected layers for the encoder and 3 fully-connected layers for the decoder.[7] They were trained by the Adam optimizer with a learning rate 0.001. The other conditions were the same as those given in Section IV-A. For reference, the plain autoencoders (AEs) without probabilistic assumptions, whose network architectures were the same as VAEs, were also compared.

PESQ and STOI of the reconstructed signals are shown in Figs. 10 and 11, respectively. To compare the different networks in a single figure, the scores are plotted w.r.t. the number of trainable parameters, where the horizontal axis is represented with the prefix kilo (k). Note that the scores for the proposed RBM are the same as Figs. 4 and 5. As in the figures, deeper networks resulted in poor scores compared to the shallower networks. This should be due to the simple network architecture of the encoder and decoder (fully-connected network with ReLU), which causes difficulty of training, e.g., gradient vanishing. To make the training reliable, we implemented enhanced versions of the networks and trained them for better comparison as follows.

[7]The dimensions of the encoder of `L3` were $129 \rightarrow 100 \rightarrow 100 \rightarrow 100$, where the arrow represents the combination of the layers. Those of the decoder were reversed, i.e, $100 \rightarrow 100 \rightarrow 100 \rightarrow 129$. Note that, for VAEs, the output dimension of the last layer of the encoder was doubled because the output of VAEs were the 100-dimensional mean and variance vectors.

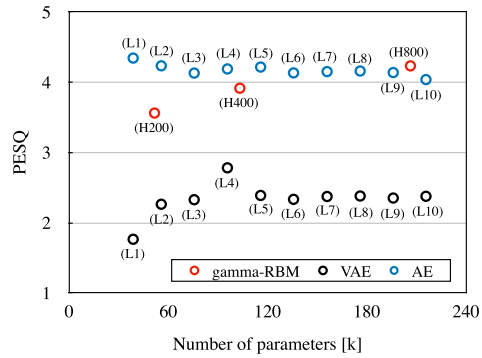

Fig. 12. PESQ scores for the proposed RBMs (red) and the enhanced AEs/VAEs (blue and black, respectively) w.r.t. the number of parameters.

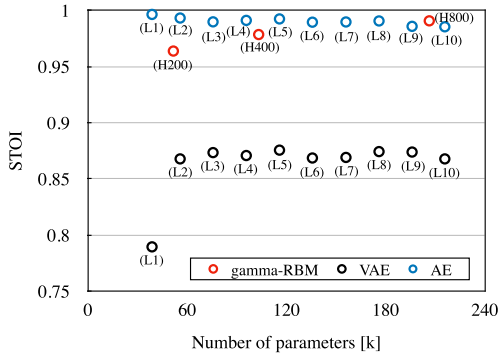

Fig. 13. STOI scores for the proposed RBMs (red) and the enhanced AEs/VAEs (blue and black, respectively) w.r.t. the number of parameters.

The network architecture of AEs/VAEs were improved by incorporating deep learning techniques. Specifically, the batch normalization layer [41] and skip connection [42] were applied to all layers. PESQ and STOI for the enhanced AEs/VAEs are shown in Figs. 12 and 13, respectively. The proposed RBM with 800 hidden layers (`H800`) resulted in the scores comparable to AEs. While some shallow AEs were better than the proposed RBM, AEs cannot be used as a generative model because they do not have probabilistic background. VAEs can be used as a generative model, but their scores were considerably lower than the RBMs. One reason for this low reconstruction performance should be because of the over-smoothing of log-amplitude spectra generated by VAEs. From these results, it can be confirmed that the proposed RBM performs well even though it can be used as a generative model.

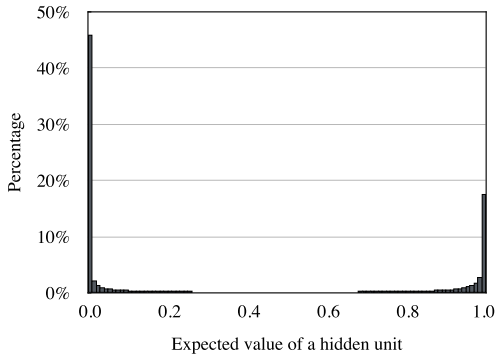

Fig. 14. Histogram of encoded values by the proposed gamma-Bernoulli RBM.

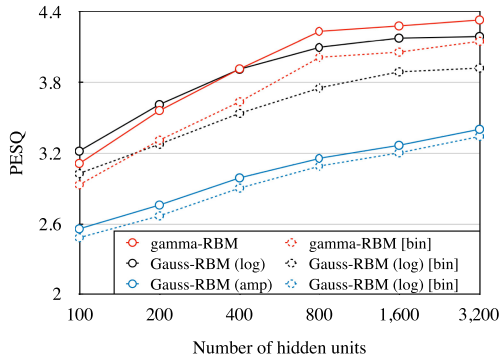

Fig. 15. PESQ scores for the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black, and `Gauss-RBM (amp)`, blue) models with (solid) and without (dotted) binarization.

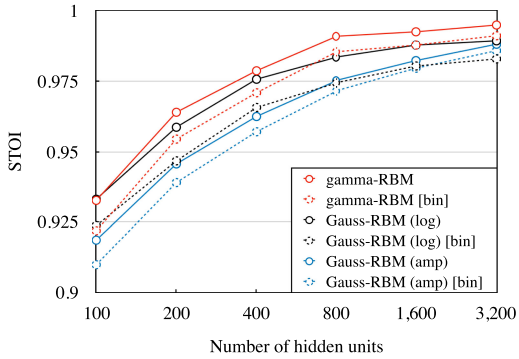

Fig. 16. STOI scores for the proposed (`gamma-RBM`, red) and conventional (`Gauss-RBM (log)`, black, and `Gauss-RBM (amp)`, blue) models with (solid) and without (dotted) binarization.

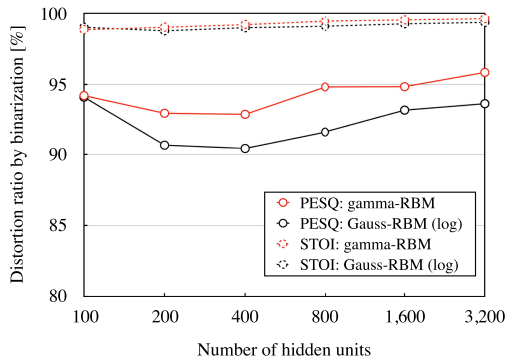

Fig. 17. Ratio of the objective scores before and after binarization.

E. Data Compression by the Binary Representation

Since the hidden units of the RBMs are assumed to be binary, the proposed RBM can provide compressive representation of data. For the above example, in the case of 800 hidden units (`H800`), a 129-dimensional vector is represented by an 800-dimensional binary vector. If the input is given by double precision (64 b), then the compression rate is $800/(129 \times 64) \approx 0.097$. That is, the proposed RBM can simultaneously extract latent features and reduce 90 % of the storage compared to the input in double precision[8]. However, since the computation inside the RBMs is performed by floating-point arithmetic, representing the latent features by strictly binary numbers requires binarization, which have not been performed in the previous subsection. Here, the effect of binarization is discussed as follows.

A histogram of the encoded values without binarization is shown in Fig. 14, where all data in the dataset were encoded by the proposed RBM (`H800`), and the number of bins was 100. As can be seen from the figure, most of the values were concentrated around 0 and 1. That is, the encoded values were approximately binary as expected from the binary assumption. The effect of binarization is caused by the values greater than 0 and less than 1, which infrequently occur according to the histogram in Fig. 14.

The effects of binarization on the reconstructed signals in terms of PESQ and STOI are shown in Figs. 15 and 16, respectively. Although binarization degraded the performance for all situations, the amount of degradation was not so significant. As STOI of the proposed RBMs *with* binarization was better than than the conventional RBMs *without* binarization for `H800` and `H3200`, the proposed RBM seems more robust against binarization. To see this quantitatively, ratio of the scores before and after binarization is shown in Fig. 17. It is calculated by dividing the dotted lines by the solid lines in Figs. 15 and 16. As in the figure, the performance of the proposed RBM did not degrade much compared to the conventional RBM. Therefore, the proposed RBM is better suited for data compression than the conventional RBM.

[8]Note that this is lossy compression because the RBMs have loss of information as shown in the experiments. Therefore, this compression rate is not so impressive compared to other lossy methods specialized for data compression. Our motivation of using RBMs is not compression but representation of the data, i.e., RBMs can extract some meaningful latent features (e.g., phonological distinctive features [43]). Hence, we do not discuss about the practical applicability of RBMs to data compression.

F. Balance Between the Linear and Logarithmic Scales

In the proposed gamma-Bernoulli RBM defined in Eq. (24), terms related to the linear scales and those related to the logarithmic scales are summed together. One may suspect the optimality of this model: it might be possible to obtain a better model by adjusting the effect of those two kinds of terms. For example, one may emphasize the contribution from the logarithmic scale more than that from the linear scale so that the gamma-Bernoulli RBM can perform better for audio signals. To investigate the

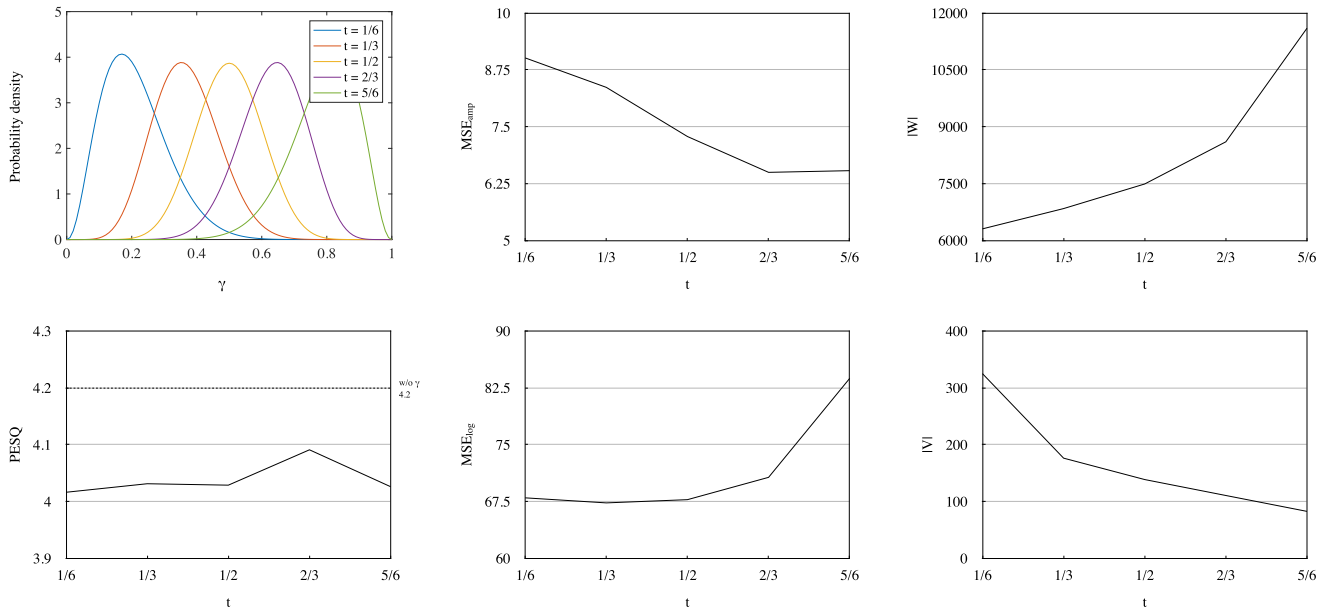

Fig. 18. Result of the experiment investigating the balance of linear and logarithmic scales (Section IV-F). Since the hyperparameter $\boldsymbol{\gamma}$ is controlled by the scalar parameter t, their relation is illustrated in the top-left figure as PDFs of the prior distributions of $\boldsymbol{\gamma}$ (the variance parameter s was fixed to 0.01). The other five figures illustrate the result: from bottom left to bottom right, PESQ, $\mathrm{MSE}_{\mathrm{amp}}$, $\mathrm{MSE}_{\mathrm{log}}$, $\|\mathbf{W}\|_F$, and $\|\mathbf{V}\|_F$.

correctness of this expectation, we propose an extension of the gamma-Bernoulli RBM that can emphasize the contribution from one of the two scales.

Since this section is for experimental investigations, details of the proposed extension are separately given in Appendix A. Here, we briefly summarize the features of the extended RBM. The gamma-Bernoulli RBM is extended by a trainable weight $\boldsymbol{\gamma} \in [0, 1]^D$ that can emphasize one of the two scales: only linear scale is considered when $\boldsymbol{\gamma} \to \mathbf{0}$, and only logarithmic scale is considered when $\boldsymbol{\gamma} \to \mathbf{1}$. It can take values between $\mathbf{0}$ and $\mathbf{1}$, and hence it can balance the contributions from the two scales. For easier adjustment of the weight, $\boldsymbol{\gamma}$ is associated with a scalar t such that $t \to 0$ prefers the linear scale, and $t \to 1$ prefers the logarithmic scale.

By changing the parameter t that controls the contributions from the two scales, we investigated the effect of balancing the scales. The results are illustrated in Fig. 18, where the number of the hidden units was 800. From the bottom-left figure, it can be seen that PESQ scores did not vary much by changing t. Note that the proposed gamma-Bernoulli RBM without the extension achieved 4.23 with the same number of hidden units (see Fig. 4, `H800`), i.e., the extension resulted in worse PESQ scores. Therefore, the gamma-Bernoulli RBM proposed in Section III-C is a reasonable model for handling audio signals with consideration of both linear and logarithmic scales. Interestingly, the other four metrics show trade-off relations: $\mathrm{MSE}_{\mathrm{amp}}$ and $\mathrm{MSE}_{\mathrm{log}}$ were traded by t, and likewise, $\|\mathbf{W}\|_F$ and $\|\mathbf{V}\|_F$ were traded by t. These trade-off relations indicate that the effect of t (and hence $\boldsymbol{\gamma}$) was absorbed by $\mathbf{W}$ and $\mathbf{V}$. In other words, the proposed gamma-Bernoulli RBM can automatically balance the importance of the two scales by adjusting the magnitude of $\mathbf{W}$ and $\mathbf{V}$.

TABLE VI
PESQ OF GAMMA-BERNOULLI RBM AND GAMMA-GAMMA RBM

Number of hidden units	H50	H100	H200	H400	H800
gamma-gamma RBM	**3.50**	**3.91**	**4.21**	**4.24**	4.23
gamma-Bernoulli RBM	2.64	3.11	3.56	3.91	4.23

G. Gamma-Gamma RBM

From the viewpoint of information compression, considering the binary hidden units is reasonable. Yet, an upper bound of the expressive power of an RBM based on the gamma distribution is also interesting to investigate because, to the best of our knowledge, such investigation has not been performed in the literature. To investigate the expressive power of a gamma-distribution-based RBM, we propose an extension of the gamma-Bernoulli RBM that handles not only visible but also hidden units by the gamma distribution.

Details of the extended RBM, named *gamma-gamma RBM*, are given in Appendix B. Here, its features are briefly summarized. The gamma-gamma RBM is obtained by simplifying the energy function of the gamma-Bernoulli RBM (by setting $\boldsymbol{b} = \mathbf{0}$, $\boldsymbol{c} = \mathbf{0}$, $\boldsymbol{d} = -\mathbf{1}$, and omitting the exponential function of the hidden units). Then, the two conditional distributions, $p(\boldsymbol{v}|\boldsymbol{h})$ and $p(\boldsymbol{h}|\boldsymbol{v})$, are represented by the gamma distribution. Similarly to the gamma-Bernoulli RBM, the constraints on the parameters of the gamma distribution must also be satisfied by the gamma-gamma RBM. This is realized by division using a small positive constant ε which was set to 10^{-10} in this experiment. Optimization of the parameters was performed in the same manner as for the gamma-Bernoulli RBM.

PESQ scores of both the gamma-gamma RBM and gamma-Bernoulli RBM are shown in Table 6. Since the gamma-gamma

RBM treats the hidden units as positive real numbers, its expressive power should be higher than the gamma-Bernoulli RBM that treats the hidden units as binary numbers. Such tendency can be clearly seen when the number of hidden units was small (i.e., less than or equals to 400). Therefore, the gamma-gamma RBM should be preferable if the number of hidden units is forced to be small. In such a case, information compression occurs owing to the smallness of the number of hidden units (especially when the number is smaller than the dimension of the data vectors). However, difference of the performances between the gamma-gamma RBM and gamma-Bernoulli RBM became small for the larger number of hidden units (i.e., more than or equals to 800). Thus, the gamma-Bernoulli RBM is more interesting than the gamma-gamma RBM in terms of information compression when the number of hidden units can be sufficiently large.

V. Conclusion

In this paper, we proposed a novel RBM named gamma-Bernoulli RBM. At first, we introduced a general gamma Boltzmann machine and showed that its conditional distribution is given by the gamma distribution. Then, we proposed the gamma-Bernoulli RBM by restricting the general gamma Boltzmann machine. Since its conditional distributions are given by the gamma and Bernoulli distributions, its training is practically tractable as the ordinary Gaussian-Bernoulli RBM. By modeling observable data via the gamma distribution, the proposed RBM can naturally handle positive data such as magnitude spectra. As it optimizes the parameters by considering data simultaneously in the linear and logarithmic scales, a trained gamma-Bernoulli RBM should be suitable for an application sensitive to both linear and logarithmic quantities. The properties and effectiveness of the proposed RBM were investigated through speech representation experiments, and its potential of audio modeling was demonstrated. Two extensions of the proposed RBM were also proposed for further investigation.

This paper presented the basic gamma-Bernoulli RBM and focused on investigation of its properties, because this is the first step of the research. There are many possible directions of research towards practical applications. One obvious direction is to consider a deep network based on the proposed RBM. Stacking it should be interesting for improving the ability of modeling complicated data structure. Another direction is to construct a model that handles complex-valued data. Some recent speech processing systems target raw waveform or complex spectrogram for directly handling phase information [43]–[47]. Since the proposed RBM can handle magnitude simultaneously in the linear and logarithmic domains, its combination with a system that handles phase information should be a promising approach to complex-valued data modeling. Proceeding these directions as well as developing applications of the gamma-Bernoulli RBM are left as the future works.

Appendix A
Gamma-Bernoulli RBM With Adjustable Scale

Some tasks may prefer a linear scale more than a logarithmic scale, and the other tasks may be in the opposite situation. To balance the contributions from linear and logarithmic scales, a weighting parameter $\boldsymbol{\gamma} \in [0,1]^D$ is introduced into the proposed gamma-Bernoulli RBM as follows:

$$\begin{aligned} E^{\boldsymbol{\gamma}}_{\Gamma\mathrm{B}}(\boldsymbol{v},\boldsymbol{h}) = &-(\boldsymbol{v}\circ(\mathbf{1}-\boldsymbol{\gamma}))^{\mathsf{T}}\mathbf{W}(\exp(\boldsymbol{h})-\mathbf{1}) \\ &-(\log(\boldsymbol{v})\circ\boldsymbol{\gamma})^{\mathsf{T}}(\mathbf{V}\boldsymbol{h}-\mathbf{1})-\boldsymbol{d}^{\mathsf{T}}\boldsymbol{h}, \end{aligned} \tag{42}$$

where $p(\boldsymbol{v}|\boldsymbol{h})$ becomes the exponential distribution when $\boldsymbol{\gamma}\to\mathbf{0}$ and becomes the Pareto distribution when $\boldsymbol{\gamma}\to\mathbf{1}$. Note that the parameter $\boldsymbol{c}$ is omitted based on the experiment in Section IV-B (see Table 1).

To manually adjust the hyperparameter $\boldsymbol{\gamma}$ for each task, maximum *a posteriori* (MAP) estimation is considered:

$$L_{\mathrm{MAP}}(\boldsymbol{\theta}) = \log(p(\boldsymbol{\theta}|\{\boldsymbol{v}\}_1^N)) \tag{43}$$

$$\propto \log(p(\{\boldsymbol{v}\}_1^N|\boldsymbol{\theta})) + \log(p(\boldsymbol{\theta})), \tag{44}$$

where the parameter-shared i.i.d beta distribution is applied for the prior distribution of $\boldsymbol{\gamma}$,

$$p(\boldsymbol{\gamma};\alpha,\beta) = \prod_i \mathrm{Beta}(\gamma_i;\alpha,\beta), \tag{45}$$

$$\mathrm{Beta}(x;\alpha,\beta) = \frac{x^{\alpha-1}(1-x)^{\beta-1}}{B(\alpha,\beta)}, \tag{46}$$

and $B(\alpha,\beta)$ is the beta function. To make variance of the beta distribution constant regardless of its mode (see Fig. 18), we propose the following parametrization of $\alpha(t)$ and $\beta(t)$:

$$\alpha = r_s(\tau)\sin(\tau), \qquad \beta = r_s(\tau)\cos(\tau), \tag{47}$$

where r_s is defined with a user-given variance parameter s,

$$\tau(t) = \frac{\pi t}{2}, \qquad r_s(\tau) = \frac{(1-2\tilde{}s)\sin(\tau)\cos(\tau)-s}{s\,(\sin(\tau)+\cos(\tau))^3}, \tag{48}$$

$s\in(0,1/12]$, $t\in(1/2-T(s),1/2+T(s))$ $(\subset[0,1])$, and $T(s)=(2/\pi)\tan^{-1}(\sqrt{1-4\tilde{}s})$. Smaller t prefers the linear scale, and larger t prefers the logrithmic scale.

The partial derivative of the objective function w.r.t. $\boldsymbol{\gamma}$ is

$$\frac{\partial L_{\mathrm{MAP}}}{\partial\boldsymbol{\gamma}} \approx \Big\langle -\frac{\partial E^{\boldsymbol{\gamma}}_{\Gamma\mathrm{B}}}{\partial\boldsymbol{\gamma}}\Big\rangle_{\mathrm{data}} - \Big\langle -\frac{\partial E^{\boldsymbol{\gamma}}_{\Gamma\mathrm{B}}}{\partial\boldsymbol{\gamma}}\Big\rangle_{\mathrm{recon}} + \frac{\alpha-1}{\boldsymbol{\gamma}} - \frac{\beta-1}{\mathbf{1}-\boldsymbol{\gamma}}, \tag{49}$$

where the partial derivative of the energy function is given as

$$-\frac{\partial E^{\boldsymbol{\gamma}}_{\Gamma\mathrm{B}}}{\partial\boldsymbol{\gamma}} = \log(\boldsymbol{v})\circ(\mathbf{V}\boldsymbol{h}-\mathbf{1}) - \boldsymbol{v}\circ(\mathbf{W}(\exp(\boldsymbol{h})-\mathbf{1})). \tag{50}$$

By parametrizing $\boldsymbol{\gamma}$ to satisfy $\mathbf{0}<\boldsymbol{\gamma}<\mathbf{1}$ as

$$\boldsymbol{\gamma} = f_\sigma(\widetilde{\boldsymbol{\gamma}}), \tag{51}$$

the hyperparameter $\widetilde{\boldsymbol{\gamma}}$ can be optimized using the chain rule:

$$\frac{\partial L_{\mathrm{MAP}}}{\partial\widetilde{\boldsymbol{\gamma}}} = \frac{\partial\boldsymbol{\gamma}}{\partial\widetilde{\boldsymbol{\gamma}}}\circ\frac{\partial L_{\mathrm{MAP}}}{\partial\boldsymbol{\gamma}} = \boldsymbol{\gamma}\circ(\mathbf{1}-\boldsymbol{\gamma})\circ\frac{\partial L_{\mathrm{MAP}}}{\partial\boldsymbol{\gamma}}, \tag{52}$$

For the other parameters, the uniform distribution is considered as the prior. Then, by recasting the visible units as

$$\boldsymbol{v}\to\boldsymbol{v}\circ(\mathbf{1}-\boldsymbol{\gamma}), \qquad \log(\boldsymbol{v})\to\log(\boldsymbol{v})\circ\boldsymbol{\gamma}, \tag{53}$$

gradients w.r.t $\widetilde{\mathbf{W}}$ and $\widetilde{\mathbf{V}}$ stay the same as in Eqs. (35)–(38).

APPENDIX B
GAMMA-GAMMA RBM

While the binary nature of hidden units should be more interesting in terms of data representation, we can define a gamma-gamma RBM that handles hidden units as positive numbers. It is defined via the following energy function:

$$E_{\Gamma\Gamma}(\boldsymbol{v},\boldsymbol{h}) = -\boldsymbol{v}^{\mathsf{T}}\mathbf{W}\boldsymbol{h} - \log(\boldsymbol{v})^{\mathsf{T}}\mathbf{V}\log(\boldsymbol{h}) \tag{54}$$

$$+ \log(\boldsymbol{v})^{\mathsf{T}}\mathbf{1} + \log(\boldsymbol{h})^{\mathsf{T}}\mathbf{1}, \tag{55}$$

that can be derived from Eq. (12) by inserting

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v}\\ \boldsymbol{h}\end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix}\mathbf{O} & \mathbf{W}\\ \mathbf{W}^{\mathsf{T}} & \mathbf{O}\end{bmatrix}, \qquad \boldsymbol{u} = \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \tag{56}$$

$$\mathbf{S} = \begin{bmatrix}\mathbf{O} & \mathbf{V}\\ \mathbf{V}^{\mathsf{T}} & \mathbf{O}\end{bmatrix}, \qquad \boldsymbol{s} = \begin{bmatrix}-\mathbf{1}\\ -\mathbf{1}\end{bmatrix}. \tag{57}$$

where $-\mathbf{W} \in \mathbb{R}_{++}^{D\times H}$, and $\mathbf{V} \in \mathbb{R}_{++}^{D\times H}$. This energy function defines an RBM with the following conditional distributions:

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{G}(\boldsymbol{v}; \mathbf{V}\log(\boldsymbol{h}), -\mathbf{W}\boldsymbol{h}), \tag{58}$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \mathcal{G}(\boldsymbol{h}; \mathbf{V}^{\mathsf{T}}\log(\boldsymbol{v}), -\mathbf{W}^{\mathsf{T}}\boldsymbol{v}), \tag{59}$$

i.e., both visible and hidden units are handled by the gamma distribution, and hence gamma-gamma RBM.

In order to fulfill the condition of the gamma distribution, $\mathbf{V}\log(\boldsymbol{h}) > \mathbf{0}$ and $\mathbf{V}^{\mathsf{T}}\log(\boldsymbol{v}) > \mathbf{0}$ must be satisfied. These inequalities depend on $\boldsymbol{h}$ and $\boldsymbol{v}$, and therefore they must be modified. By setting a small positive constant $\varepsilon > 0$ such that $h_i > \varepsilon$ and $v_i > \varepsilon$ for all i, we redefine $\boldsymbol{s}$ as

$$\boldsymbol{s} = \begin{bmatrix}-\mathbf{1} - \log(\varepsilon)\mathbf{V}\mathbf{1}\\ -\mathbf{1} - \log(\varepsilon)\mathbf{V}^{\mathsf{T}}\mathbf{1}\end{bmatrix}. \tag{60}$$

Then, the corresponding conditional distributions become

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{G}(\boldsymbol{v}; \mathbf{V}\log(\boldsymbol{h}/\varepsilon), -\mathbf{W}\boldsymbol{h}), \tag{61}$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \mathcal{G}(\boldsymbol{h}; \mathbf{V}^{\mathsf{T}}\log(\boldsymbol{v}/\varepsilon), -\mathbf{W}^{\mathsf{T}}\boldsymbol{v}), \tag{62}$$

which satisfy $\mathbf{V}\log(\boldsymbol{h}/\varepsilon) > \mathbf{0}$ and $\mathbf{V}^{\mathsf{T}}\log(\boldsymbol{v}/\varepsilon) > \mathbf{0}$ whenever $h_i, v_i > \varepsilon\ \forall i$.

ACKNOWLEDGMENT

The authors would like to thank Dr. Ryo Karakida and Dr. Sho Sonoda for their valuable comments. The authors also thank the anonymous reviewers for their valuable comments.

REFERENCES

[1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013, *arXiv:1312.6114*.
[2] L. Pandey, A. Kumar, and V. Namboodiri, "Monoaural audio source separation using variational autoencoders," in *Proc. Interspeech*, 2018, pp. 3489–3493.
[3] M. Sadeghi, S. Leglaive, X. Alameda-Pineda, L. Girin, and R. Horaud, "Audio-visual speech enhancement using conditional variational auto-encoders," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1788–1800, 2020.
[4] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[5] Y. Saito, S. Takamichi, and H. Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, pp. 84–96, Jan. 2017.
[6] M. H. Soni, N. Shah, and H. A. Patil, "Time-frequency masking-based speech enhancement using generative adversarial network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2018, pp. 5039–5043.
[7] K. Kumar *et al.*, "MelGAN: Generative adversarial networks for conditional waveform synthesis," in *Proc. Adv. Neural Inform. Process. Syst.*, 2019, pp. 14 910–14 921.
[8] A. v. d. Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proc. ICML*, 2016, pp. 1747–1756..
[9] A. v. d. Oord *et al.*, "WaveNet: A generative model for raw audio," 2016, *arXiv:1609.03499*.
[10] D. J. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. ICML'15: Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, 2015, pp. 1530–1538.
[11] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 3617–3621.
[12] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for Boltzmann machines," *Cogn. Sci.*, vol. 9, no. 1, pp. 147–169, 1985.
[13] Y. Freund and D. Haussler, "Unsupervised learning of distributions of binary vectors using two layer networks," in *Proc. NIPS*, 1994, pp. 912–919,.
[14] H. Lee, C. Ekanadham, and A. Y. Ng, "Sparse deep belief net model for visual area V2," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 873–880.
[15] R. Salakhutdinov, A. Mnih, and G. Hinton, "Restricted Boltzmann machines for collaborative filtering," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 791–798.
[16] Y. Chen, L. Lu, and X. Li, "Application of continuous restricted Boltzmann machine to identify multivariate geochemical anomaly," *J. Geochemical Exploration*, vol. 140, pp. 56–63, 2014.
[17] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.
[18] K. Cho, A. Ilin, and T. Raiko, "Improved learning of Gaussian-Bernoulli restricted Boltzmann machines," in *Proc. Internet Corporation Assigned Names Numbers*. 2011, pp. 10–17.
[19] A.-R. Mohamed and G. Hinton, "Phone recognition using restricted Boltzmann machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2010, pp. 4354–4357.
[20] Z. Oo, Y. Kawakami, L. Wang, S. Nakagawa, X. Xiao, and M. Iwahashi, "DNN-based amplitude and phase feature enhancement for noise robust speaker identification," in *Proc. Interspeech*, 2016, pp. 2204–2208.
[21] Y.-J. Hu and Z.-H. Ling, "Dbn-based spectral feature representation for statistical parametric speech synthesis," *IEEE Signal Process. Lett.*, vol. 23, no. 3, pp. 321–325, Mar. 2016.
[22] T. Nakashika, T. Takiguchi, and Y. Minami, "Non-parallel training in voice conversion using an adaptive restricted boltzmann machine," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 11, pp. 2032–2045, Nov. 2016.
[23] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "Tandem-Straight: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, F0, and aperiodicity estimation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2008, pp. 3933–3936.
[24] M. Li, Z. Miao, and C. Ma, "Feature extraction with convolutional restricted boltzmann machine for audio classification," in *Proc. 3rd IAPR Asian Conf. Pattern Recognit.*, 2015, pp. 791–795.
[25] Z.-H. Ling, L. Deng, and D. Yu, "Modeling spectral envelopes using restricted boltzmann machines for statistical parametric speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2013, pp. 7825–7829.
[26] T. Nakashika, "LSTBM: A novel sequence representation of speech spectra using restricted boltzmann machine with long short-term memory," in *Proc. Interspeech*, 2018, pp. 2529–2533.
[27] N. Jaitly and G. Hinton, "Learning a better representation of speech soundwaves using restricted boltzmann machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2011, pp. 5884–5887.
[28] H. B. Sailor and H. A. Patil, "Novel unsupervised auditory filterbank learning using convolutional RBM for speech recognition," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 12, pp. 2341–2353, 2016.
[29] T. Nakashika and K. Yatabe, "Gamma boltzmann machine for simultaneously modeling linear- and log-amplitude spectra," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, 2020, pp. 471–476.

[30] S. Vasquez and M. Lewis, "MelNet: A generative model for audio in the frequency domain," 2019, *arXiv:1906.01083*.
[31] Y. Wang *et al.*, "Tacotron: Towards End-to-End Speech Synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.
[32] T. Kaneko, S. Takaki, H. Kameoka, and J. Yamagishi, "Generative adversarial network-based postfilter for STFT spectrograms," in *Proc. Interspeech*, 2017, pp. 3389–3393.
[33] Z. Zhao, L. Guo, M. Jia, and L. Wang, "The generalized gamma-DBN for high-resolution SAR image classification," *Remote Sens.*, vol. 10, no. 6, pp. 1–24, 2018.
[34] S. Gazor and W. Zhang, "Speech probability distribution," *IEEE Signal Process. Lett.*, vol. 10, no. 7, pp. 204–207, 2003.
[35] J. W. Shin, J.-H. Chang, and N. S. Kim, "Statistical modeling of speech signals based on generalized gamma distribution," *IEEE Signal Process. Lett.*, vol. 12, no. 3, pp. 258–261, Mar. 2005.
[36] R. Martin, "Speech enhancement using MMSE short time spectral estimation with gamma distributed speech priors," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Process.*, vol. 1, 2002, pp. 253–256.
[37] I. Andrianakis and P. R. White, "MMSE speech spectral amplitude estimators with chi and gamma speech priors," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. Proc.*, vol. 3, 2006, pp. 1068–1071p. III.
[38] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[39] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–15.
[40] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoder-based high-quality speech synthesis system for real-time applications," *IEICE Trans. Inf. Syst.*, vol. 99, no. 7, pp. 1877–1884, 2016.
[41] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*. PMLR, 2015, pp. 448–456.
[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
[43] S.-W. Fu, Y. Tsao, X. Lu, and H. Kawai, "Raw waveform-based speech enhancement by fully convolutional networks," in *Proc. Asia-Pacific Signal, Inf. Process. Assoc. Annu. Summit Conf.*, 2017, pp. 006–012.
[44] S.-W. Fu, T.-y. Hu, Y. Tsao, and X. Lu, "Complex spectrogram enhancement by convolutional neural network with multi-metrics learning," in *Proc. Mach. Learn. Signal Process. Tech. Committee*, 2017, pp. 1–6.
[45] N. Zeghidour, N. Usunier, G. Synnaeve, R. Collobert, and E. Dupoux, "End-to-end speech recognition from the raw waveform," in *Proc. Interspeech*, 2018, pp. 781–785..
[46] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *Proc. Spoken Lang. Technol. Workshop*, 2018, pp. 1021–1028.
[47] D. S. Williamson, Y. Wang, and D. Wang, "Complex ratio masking for monaural speech separation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 3, pp. 483–492, 2015.

Toru Nakashika (Member, IEEE) received the B.E. and M.E. degrees in computer science from Kobe University, in 2009 and 2011, respectively. On the summer in 2010, he was a Student Researcher with IBM Research, Tokyo Research Laboratory. From 2011 to 2012, he was a Visiting Researcher in the image group with INSA de Lyon in France. In the same year, he continued his research as a Doctoral Student with Kobe University, and received his Dr.Eng. degree in computer science in 2014. Till April 2015, he was an Assistant Professor with Kobe University. In 2015, he joined the University of Electro-Communications as an Assistant Professor. He is currently an Associate Professor of the Graduate School of Informatics and Engineering, the University of Electro-Communications. He was the recipient of the Young Researcher's Award in IEICE Speech Field in 2013, the Best Paper Award in SIGMUS Ongaku Symposium 2016, the 44th Awaya Prize Young Researcher Award from the Acoustical Society of Japan, the 15th Itakura Prize Innovative Young Researcher Award from the Acoustical Society of Japan. He is a Member of the IEEE, the IEICE, the ASJ, the JSAI, and the ISCA.

Kohei Yatabe (Member, IEEE) received the B.E., M.E., and Ph.D. degrees from Waseda University in 2012, 2014, and 2017, respectively. He is currently an Assistant Professor with the Department of Intermedia Art and Science, Waseda University.