Interspeech 2018
2-6 September 2018, Hyderabad

LSTBM: A Novel Sequence Representation of Speech Spectra Using Restricted Boltzmann Machine with Long Short-Term Memory

Toru Nakashika[1]

[1]The University of Electro-Communications

nakashika@uec.ac.jp

Abstract

In this paper, we propose a novel probabilistic model, namely long short-term Boltzmann memory (LSTBM), to represent sequential data like speech spectra. The LSTBM is an extension of a restricted Boltzmann machine (RBM) that has generative long short-term memory (LSTM) units. The original RBM automatically learns relationships between visible and hidden units and is widely used as a feature extractor, a generator, a classifier, a pre-training method of deep neural networks, etc. However, the RBM is not sufficient to represent sequential data because it assumes that each frame from sequential data is completely independent of the others. Unlike conventional RBMs, the LSTBM has connections over time via LSTM units and represents time dependencies in sequential data. Our speech coding experiments demonstrated that the proposed LSTBM outperformed the other conventional methods: an RBM and a temporal RBM.

Index Terms: restricted Boltzmann machine, long short-term memory, neural networks, speech parameterization, speech synthesis, sequence modeling

1. Introduction

Deep learning is one of the recent hottest topics in wide research fields such as artificial intelligence, machine learning, and signal processing including image classification, speech recognition, etc [1]. Most of the deep learning methods are based on deterministic approaches of neural networks that learn transfer functions from inputs to outputs while some approaches based on generative models, such as variational auto-encoders (VAEs) [2] and generative adversarial networks (GANs) [3], have been garnering much attentions recent years. One of the other most powerful generative models is restricted Boltzmann machines (RBMs) [4, 5, 6]. The RBM is a probabilistic model that defines relations between visible and hidden units with an energy function and has often been used as a feature extractor, a generator, a classifier, and a pre-training scheme of deep neural networks (DNNs) [5]. Many extensions and variations of the RBM have been proposed so far [7, 8, 9, 10, 11]. Although RBMs have been used in so many tasks, they traditionally assumed the frame-independencies even for sequential data.

On the other hand, in speech signal processing, image processing, and natural language processing, in which we need to deal with sequential data, recurrent neural networks (RNNs), convolutional neural networks (CNNs), long short-term memory (LSTM) networks [12], and gated recurrent unit (GRU) networks [13] are often effective due to the ability to capture temporal dependencies. Among them, the LSTM can convey important and characteristic information through time with its "memory cells" inside, and it has been reported that the LSTM drastically improved the performance of speech recognition [14], speech synthesis [15], and machine translation [16] compared to the simple RNNs. The GRU also has similar functions and produces similar results to those of the LSTM [17]; however, the GRU does not assume the existence of memory cells explicitly.

The RBM is not sufficient to represent sequential data because each frame is assumed independent of the other frames while the dependencies through time should exist. Some extensions of the model, temporal RBM (TRBM) [18] and recurrent temporal RBM (RTRBM) [19], have been also proposed to deal with sequential data. However, these methods include several problems, such as vanishing gradient, exploding gradient, and difficulties to capture long term dependencies, as well as in the simple RNNs. In this paper, we propose and evaluate an extension of the RBM that imitates the functions of memory units to those of the LSTM networks as a generative model, called long short-term Boltzmann memory (LSTBM). The LSTBM is defined as a probabilistic model that consists of visible units and different sets of hidden units (input gate, output gate, forget gate, and memory core units), and these hidden units determine memory cells and hidden states at the current time and convey the memory to the following time. Another characteristic of the LSTBM is that the model can be trained only using the input data under the maximum likelihood (ML) estimation; it is also possible to use the stacked LSTBMs for pre-training very deep LSTM networks like deep belief networks (DBNs) using RBMs [5]. The LSTBM also differs from LSTM-RTRBM [20] in terms of being a complete generative model.

This paper is organized as follows: in Section 2, we overview the conventional RBM. In Section 3, we define the proposed LSTBM and show its parameter estimation algorithm. In Section 4, we show our experimental results and conclude our findings in Section 5.

2. Preliminaries

In this section, we briefly recall two related models: a restricted Boltzmann machine (RBM) and its temporal extension, a temporal RBM (TRBM).

2.1. RBM

A restricted Boltzmann machine (RBM [5, 21]), one of the most-widely used energy-based models, is convenient for representing latent features that cannot be observed but surely exist in the background. An RBM was originally introduced as an undirected graphical model that defines the distribution of binary visible variables with binary hidden (latent) variables and was later extended to deal with real-valued data [5, 21] and even complex-valued data [11]. In the remaining of this paper, we refer to the real-valued version (Gaussian-Bernoulli RBM) just as an RBM. In the modeling using an RBM, the joint probability $p(\boldsymbol{v}, \boldsymbol{h})$ of real-valued visible units $\boldsymbol{v} \in \mathbb{R}^I$ and binary-valued hidden units $\boldsymbol{h} \in \mathbb{B}^J$ (I and J are the numbers of dimensions

10.21437/Interspeech.2018-1753

in the visible and hidden units, respectively, and $\mathbb{B}$ indicates the binary space that takes the value of either 0 or 1) is defined as follows:

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

$$E(\boldsymbol{v}, \boldsymbol{h}) = \frac{1}{2} \| \frac{\boldsymbol{v} - \boldsymbol{b}}{\boldsymbol{\sigma}} \|_2^2 - \boldsymbol{c}^\top \boldsymbol{h} - (\frac{\boldsymbol{v}}{\boldsymbol{\sigma}^2})^\top \mathbf{W} \boldsymbol{h} \tag{2}$$

$$Z = \int \sum_{\boldsymbol{h}} e^{-E(\boldsymbol{v}, \boldsymbol{h})} d\boldsymbol{v} \tag{3}$$

where $\boldsymbol{\theta} = \{\boldsymbol{b}, \boldsymbol{c}, \mathbf{W}, \boldsymbol{\sigma}\}$ indicates a set of parameters, which contains bias parameters of the visible units $\boldsymbol{b} \in \mathbb{R}^I$, bias parameters of the hidden units $\boldsymbol{c} \in \mathbb{R}^J$, the connection weight parameters between visible-hidden units $\mathbf{W} \in \mathbb{R}^{I \times J}$, and the standard deviation parameters $\boldsymbol{\sigma} \in \mathbb{R}^I$. The fraction bar $\frac{\cdot}{\cdot}$ and the square $\cdot^2$ indicate element-wise division and element-wise square operation, respectively. From the above definition, the conditional probabilities $p(\boldsymbol{v}|\boldsymbol{h})$ and $p(\boldsymbol{h}|\boldsymbol{v})$ form simple distributions as:

$$p(\boldsymbol{v}|\boldsymbol{h}) = \mathcal{N}(\boldsymbol{v}; \mathbf{W}\boldsymbol{h} + \boldsymbol{b}, \Delta(\boldsymbol{\sigma}^2)) \tag{4}$$

$$p(\boldsymbol{h}|\boldsymbol{v}) = \mathcal{B}(\boldsymbol{h}; \boldsymbol{\rho}(\mathbf{W}^\top (\frac{\boldsymbol{v}}{\boldsymbol{\sigma}^2}) + \boldsymbol{c}) \tag{5}$$

where $\mathcal{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\mathcal{B}(\cdot; \boldsymbol{\pi})$, $\boldsymbol{\rho}(\cdot)$, and $\Delta(\cdot)$ indicate the multivariate Gaussian distribution with the mean $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$, the multi-dimensional Bernoulli distribution with the success probabilities $\boldsymbol{\pi}$, the element-wise sigmoid function, and the function that returns a diagonal matrix whose diagonal vector is the argument, respectively.

2.2. TRBM

The temporal RBM (TRBM), proposed by Sutskever *et al.* [18], extends an RBM to represent sequential data by adding temporal connections in adjacent visible units or hidden units. Letting $\boldsymbol{x}_{A:B} = (\boldsymbol{x}_A, \boldsymbol{x}_{A+1}, \cdots, \boldsymbol{x}_B)$ denote the sequence in the interval $[A, B]$, the TRBM that assumes Markov process for hidden units defines the joint probability of the T-frame visible sequence $\boldsymbol{v}_{1:T}$ and hidden sequence $\boldsymbol{h}_{1:T}$ as

$$p(\boldsymbol{v}_{1:T}, \boldsymbol{h}_{1:T}) = \prod_{t=1}^{T} p(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1}), \tag{6}$$

which indicates that the variables $\boldsymbol{v}_t$ and $\boldsymbol{h}_t$ at the frame t only depend on the previous hidden units $\boldsymbol{h}_{t-1}$. Like an RBM, the conditional probabilities $p(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1})$ are defined as:

$$p(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1}) = \frac{1}{Z(\boldsymbol{h}_{t-1})} e^{-E(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1})} \tag{7}$$

$$\begin{aligned} E(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1}) = & \frac{1}{2} \| \frac{\boldsymbol{v}_t - \boldsymbol{b}}{\boldsymbol{\sigma}} \|_2^2 - \boldsymbol{c}^\top \boldsymbol{h}_t \\ & - (\frac{\boldsymbol{v}_t}{\boldsymbol{\sigma}^2})^\top \mathbf{W} \boldsymbol{h}_t - \boldsymbol{h}_{t-1}^\top \mathbf{U} \boldsymbol{h}_t \end{aligned} \tag{8}$$

$$Z(\boldsymbol{h}_{t-1}) = \int \sum_{\boldsymbol{h}_t} e^{-E(\boldsymbol{v}_t, \boldsymbol{h}_t | \boldsymbol{h}_{t-1})} d\boldsymbol{v}_t, \tag{9}$$

where $\mathbf{U} \in \mathbb{R}^{J \times J}$ indicates the connection weight parameters from the previous hidden units to the current units. Comparing the energy functions defined in Eqs. (2) and (8), we can notice that the TRBM includes the recurrent term of hidden units $-\boldsymbol{h}_{t-1}^\top \mathbf{U} \boldsymbol{h}_t$ while the RBM does not.

3. Proposed model: LSTBM

Even though the previously-mentioned TRBM can represent time-related dependencies with the connections from the past hidden units to the current, the information degrades by passing through time. In this paper, we propose a new probabilistic model, called long short-term Boltzmann memory (LSTBM), to tackle the feed-forward degrading problem occurring in TRBM. Introducing memory cells, the LSTBM is able to maintain inner states through time without it degrading like long short-term memory (LSTM) networks [12]. In LSTBM as well as LSTM, the J memory cells $\boldsymbol{c}_t \in \mathbb{R}^J$ at time t are calculated by adding core memory $\boldsymbol{g}_t \in \mathbb{B}^J$ passed with input gates $\boldsymbol{i}_t \in \mathbb{B}^J$ and the previous memory cells $\boldsymbol{c}_{t-1}$ passed with forget gates $\boldsymbol{f}_t \in \mathbb{B}^J$ as follows:

$$\boldsymbol{c}_t \triangleq \boldsymbol{c}_{t-1} \circ \boldsymbol{f}_t + \boldsymbol{g}_t \circ \boldsymbol{i}_t, \tag{10}$$

where $\circ$ indicates element-wise multiplication. Furthermore, we define hidden states $\boldsymbol{h}_t \in [0, 1]^J$ that will be propagated to the following time as the activations associated with the corresponding memory cells passed with output gates $\boldsymbol{o}_t \in \mathbb{B}^J$ in just the same way as LSTM does. This is formulated as:

$$\boldsymbol{h}_t \triangleq \boldsymbol{\rho}(\boldsymbol{c}_t \circ \boldsymbol{o}_t). \tag{11}$$

Note that all the hidden gates $\boldsymbol{f}_t$, $\boldsymbol{i}_t$, $\boldsymbol{o}_t$ take the values of not real numbers but either 1 or 0 indicating that the associated cells are passed through or not, respectively. The core memory would also take real numbers; however, we simply adopt binary values for the core memory in this paper.

Given that input variables (visible units or observation) $\boldsymbol{v}_t$, the hidden gates, the core memory interact each other, the LSTBM assumes Markov process on the latent variables $\boldsymbol{s}_t = \{\boldsymbol{f}_t, \boldsymbol{i}_t, \boldsymbol{o}_t, \boldsymbol{g}_t\}$, which leads to:

$$p(\boldsymbol{v}_{1:T}, \boldsymbol{s}_{1:T}) = \prod_{t=1}^{T} p(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{s}_{t-1}) \tag{12}$$

$$= \prod_{t=1}^{T} p(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1}). \tag{13}$$

In LSTBM, the joint probability $p(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1})$ at time t as follows:

$$p(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1}) = \frac{1}{Z(\boldsymbol{h}_{t-1})} e^{-E(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1})} \tag{14}$$

$$\begin{aligned} E(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1}) = & \frac{1}{2} \| \frac{\boldsymbol{v}_t - \boldsymbol{b}_v}{\boldsymbol{\sigma}} \|_2^2 \\ & - \sum_l \boldsymbol{b}^{(l)\top} \boldsymbol{s}_t^{(l)} \\ & - \sum_l (\frac{\boldsymbol{v}_t}{\boldsymbol{\sigma}^2})^\top \mathbf{W}^{(l)} \boldsymbol{s}_t^{(l)} \\ & - \sum_l \boldsymbol{h}_{t-1}^\top \mathbf{U}^{(l)} \boldsymbol{s}_t^{(l)} \end{aligned} \tag{15}$$

$$Z(\boldsymbol{h}_{t-1}) = \int \sum_{\boldsymbol{s}_t} e^{-E(\boldsymbol{v}_t, \boldsymbol{s}_t | \boldsymbol{h}_{t-1})} d\boldsymbol{v}_t, \tag{16}$$

where $l = 1, 2, 3, 4$ is an index that indicates either of the four

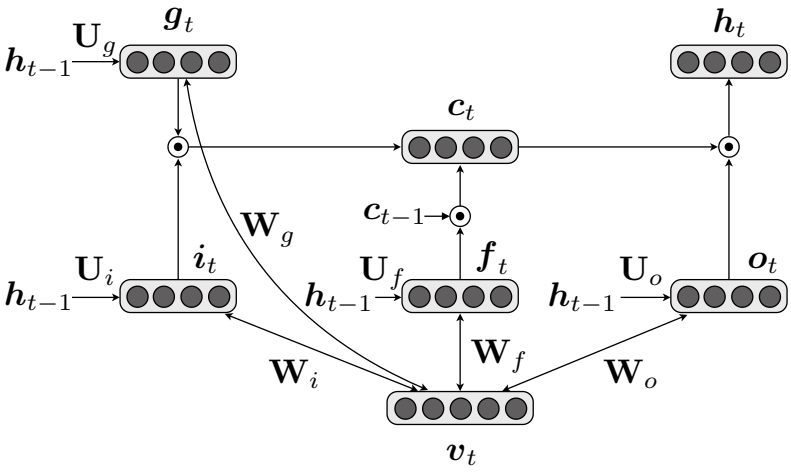

Figure 1: *Graphical representation of the extended RBM with probabilistic memory (long short-term Boltzmann memory; LSTBM).*

latent variables so that

$$(\boldsymbol{s}_t^{(l)}, \boldsymbol{b}^{(l)}, \mathbf{W}^{(l)}, \mathbf{U}^{(l)}) = \begin{cases} (\boldsymbol{f}_t, \boldsymbol{b}_f, \mathbf{W}_f, \mathbf{U}_f) & (l=1) \\ (\boldsymbol{i}_t, \boldsymbol{b}_i, \mathbf{W}_i, \mathbf{U}_i) & (l=2) \\ (\boldsymbol{o}_t, \boldsymbol{b}_o, \mathbf{W}_o, \mathbf{U}_o) & (l=3) \\ (\boldsymbol{g}_t, \boldsymbol{b}_g, \mathbf{W}_g, \mathbf{U}_g) & (l=4). \end{cases} \tag{17}$$

Here, $\boldsymbol{b}_v$, $\boldsymbol{b}^{(l)}$, $\mathbf{W}^{(l)}$, $\mathbf{U}^{(l)}$, and $\boldsymbol{\sigma}$ are bias parameters of the visible units and the latent variables, undirected connection weight parameters between the visible unit and each latent variable, recurrent connection weight parameters of each latent variable, and standard deviation parameters of the visible units, respectively. The parameters of the LSTBM $\boldsymbol{\theta} = \{\boldsymbol{b}_v, \boldsymbol{b}^{(l)}, \mathbf{W}^{(l)}, \mathbf{U}^{(l)}, \boldsymbol{\sigma} \mid \forall l\}$ can be simultaneously optimized using maximum likelihood (ML) estimation based on the gradient ascend and the contrastive divergence (CD) [5] in the way similar to an RBM and a TRBM. The LSTBM still has difficulties in inference just as a TRBM does. Therefore, we regard $\boldsymbol{h}_{t-1}$ as a constant when calculating the gradient at frame t.

From the above definition, the LSTBM can be graphically illustrated as in Figure 1. When we fix (input) the normalized visible units $\boldsymbol{v}_t$ along with the previous hidden states $\boldsymbol{h}_{t-1}$, the expected values of each latent variable can be derived as follows:

$$\mathbb{E}[\boldsymbol{f}_t] = \boldsymbol{\rho}(\mathbf{W}_f^\top \boldsymbol{v}_t + \mathbf{U}_f^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_f) \tag{18}$$
$$\mathbb{E}[\boldsymbol{i}_t] = \boldsymbol{\rho}(\mathbf{W}_i^\top \boldsymbol{v}_t + \mathbf{U}_i^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_i) \tag{19}$$
$$\mathbb{E}[\boldsymbol{o}_t] = \boldsymbol{\rho}(\mathbf{W}_o^\top \boldsymbol{v}_t + \mathbf{U}_o^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_o) \tag{20}$$
$$\mathbb{E}[\boldsymbol{g}_t] = \boldsymbol{\rho}(\mathbf{W}_g^\top \boldsymbol{v}_t + \mathbf{U}_g^\top \boldsymbol{h}_{t-1} + \boldsymbol{b}_g). \tag{21}$$

It should be noted that Eqs. (18), (19), (20), (21), (10), and (11) are equivalent to the well-know feedforward equations in the common LSTM networks. From another perspective, the proposed model can be regarded as extended representation of the traditional LSTM, and its feedforward equations are completely equivalent to that of the LSTM only when propagating the expected values of the latent variables instead of 0 or 1 values that come from the probabilities.

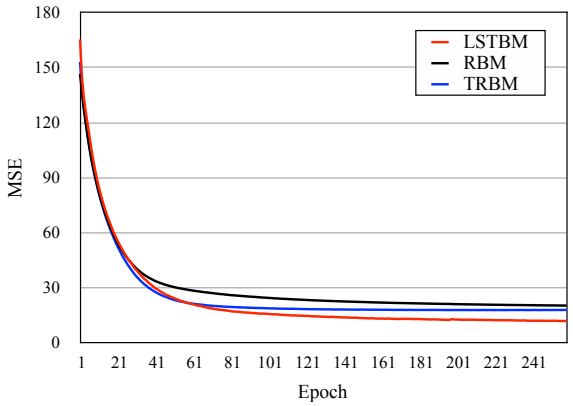

Figure 2: *MSE curves during the training.*

4. Experiments

4.1. Setup

To demonstrate the effectiveness of the proposed method, we conducted speech encoding experiments using speech signals of 50 sentences (approx. 4.2 min) for training and another 53 for test pronounced by the female announcer ("FTK") from the set "A" of the ATR speech corpora. The speech signals were downsampled from the original 20kHz to 16kHz, and processed into 129-dimensional amplitude spectra using the short-time Fourier transform (STFT) with a window length of 256 and a hop size of 64, which were set as visible units of LSTBM. The total number of frames of the training data was $64,438$. We evaluated three models of LSTBM by changing the number of hidden states to $J = 100$, 200, and 400. Each model was trained using the Adam [22] optimizer with the learning rate of 0.001, the decay rates of $\beta_1 = 0.9$ and $\beta_2 = 0.999$ in the $T = 1,000$ mini-batch and 250 epoch optimization, and compared with two conventional models: RBM and TRBM that were trained in the same conditions.

For the objective evaluation, we used PESQ (perceptual evaluation of speech quality) of the recovered signals obtained by the inverse STFT and the overlap-add method from the reconstructions of each model. The amplitude of each reconstruc-

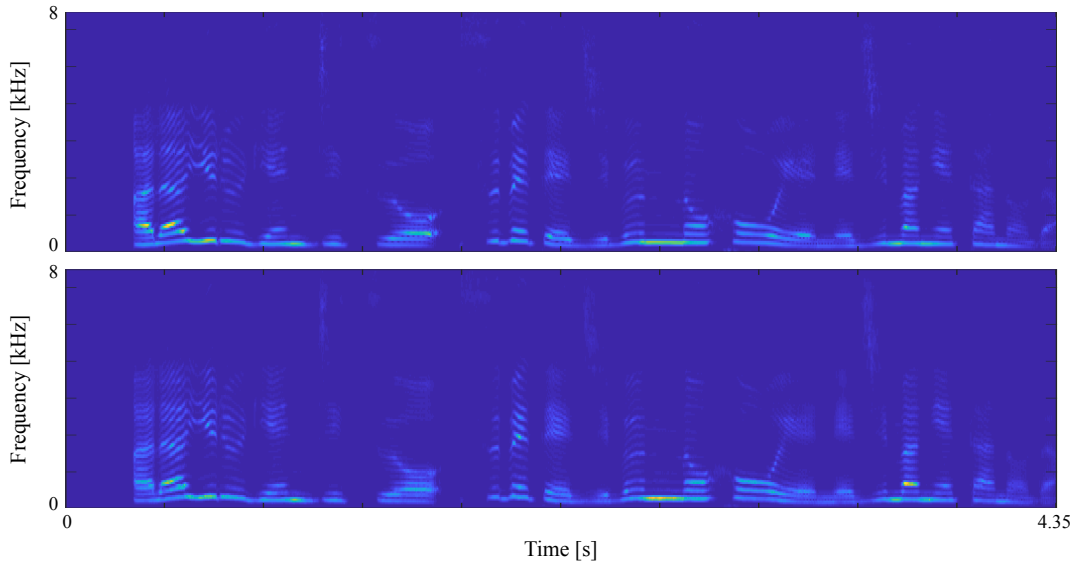

Figure 3: *Spectrograms of the original speech (above) and the reconstruction by the LSTBM (below).*

Table 1: *Encoding-decoding performance (PESQ) of the LSTBM, RBM and TRBM for training and test data. The numbers in parentheses indicate the numbers of hidden units J.*

	Train	**Test**
LSTBM(100)	3.86	3.81
LSTBM(200)	4.03	3.95
LSTBM(400)	4.11	4.02
RBM(100)	2.74	2.69
RBM(200)	3.29	3.20
RBM(400)	3.72	3.63
TRBM(100)	3.23	3.22
TRBM(200)	3.74	3.75
TRBM(400)	4.00	3.94

Table 2: *DMOS and 95% confidence intervals of each method.*

LSTBM	**RBM**	**TRBM**
4.42 ± 0.129	3.71 ± 0.152	4.36 ± 0.132

tion of complex spectra was obtained by encoding (to infer latent variables in LSTBM or hidden units in RBM and TRBM from the visible units) followed by decoding (to generate visible units from the latent variables or the hidden units), and the phases were set as the original.

4.2. Results and discussion

Figure 2 shows mean-squared error (MSE) during training of the LSTBM, the RBM and the TRBM when $J = 400$. As shown in Figure 2, the MSE of the LSTBM converged with smaller error than that of the other models. Comparing the results of the RBM and the TRBM, the TRBM converged faster than the RBM until around 100 epochs; however, both MSEs got close to each other around 250 epochs.

Table 1 summarizes the performance of each method, showing that the LSTBM outperformed the rest for training and test data when comparing with the same number of hidden units. While use of small number of hidden units ($J = 100$) degraded the performance of the RBM and the TRBM, we do not really see such degradation from the LSTBM.

We also conducted subjective evaluation using degradation mean opinion score (DMOS) listening tests. In this evaluation, nine participants listened to 20 test utterances of original speech and the reconstructed speech from each method and then selected how close the reconstructed speech sounded to the original speech on a 5-point scale (5: excellent; 4: good; 3: fair; 2: poor; and 1: bad). Each method had 400 hidden units. Table 2 shows the average DMOS and 95% confidence intervals from each method. As shown in Table 2, the LSTBM and TRBM significantly outperformed the RBM. The average DMOS of the LSTBM was slightly larger than that of the TRBM; however, there was no significant difference between them. We believe that the superiority of the LSTBM appears more significant as the number of hidden units smaller.

Finally, we observed the reconstructed spectrogram from the LSTBM (Figure 3). As shown in Figure 3, we confirm that formants and harmonics are appropriately reproduced in the reconstructed spectrogram.

5. Conclusions

In this paper, we proposed a novel energy-based generative model called long short-term Boltzmann memory (LSTBM) inspired by the temporal restricted Boltzmann machine (TRBM) and the long short-term memory (LSTM) networks. We also presented that the LSTBM feedforward terms extends the traditional LSTM networks. Experimental results showed the effectiveness of the proposed method compared to the other speech coding methods in the PSEQ criteria. We will further investigate other applications of the LSTBM such as pre-training of deep LSTM networks, etc.

6. Acknowledgements

This work was partially supported by JST ACT-I, by JSPS KAKENHI Grant Number 18K18069, and by Telecommunications Advancement Foundation Grants.

7. References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations (ICLR)*, 2014.

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014, pp. 2672–2680.

[4] Y. Freund and D. Haussler, "Unsupervised learning of distributions of binary vectors using two layer networks," 1994.

[5] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[6] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," *Computer Science Department, University of Toronto, Tech. Rep*, 2009.

[7] R. Salakhutdinov and G. E. Hinton, "Deep Boltzmann machines," *AISTATS*, pp. 448–455, 2009.

[8] A. Krizhevsky and G. E. Hinton, "Factored 3-way restricted boltzmann machines for modeling natural images," *Journal of Machine Learning Research*, 2010.

[9] K. Sohn, G. Zhou, C. Lee, and H. Lee, "Learning and Selecting Features Jointly with Point-wise Gated Boltzmann Machines," *ICML (2)*, 2013.

[10] T. Nakashika, T. Takiguchi, and Y. Minami, "Non-parallel training in voice conversion using an adaptive restricted Boltzmann machine," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2032–2045, 2016.

[11] T. Nakashika, S. Takaki, and J. Yamagishi, "Complex-valued restricted Boltzmann machine for direct learning of frequency spectra," *Proc. Interspeech 2017*, pp. 4021–4025, 2017.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *EMNLP*, 2014, pp. 1724–1734.

[14] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *ICASSP 2013*. IEEE, 2013, pp. 6645–6649.

[15] Z. Wu and S. King, "Investigating gated recurrent networks for speech synthesis," in *ICASSP 2016*. IEEE, 2016, pp. 5140–5144.

[16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[17] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *NIPS 2014 Workshop on Deep Learning*, pp. 1–9, 2014.

[18] I. Sutskever and G. Hinton, "Learning multilevel distributed representations for high-dimensional sequences," in *Artificial Intelligence and Statistics*, 2007, pp. 548–555.

[19] I. Sutskever, G. E. Hinton, and G. W. Taylor, "The recurrent temporal restricted Boltzmann machine," in *Advances in Neural Information Processing Systems*, 2009, pp. 1601–1608.

[20] Q. Lyu, Z. Wu, J. Zhu, and H. Meng, "Modelling high-dimensional sequences with LSTM-RTRBM: Application to polyphonic music generation." in *IJCAI*, 2015, pp. 4138–4139.

[21] K. Cho, A. Ilin, and T. Raiko, "Improved learning of Gaussian-Bernoulli restricted Boltzmann machines," in *Artificial Neural Networks and Machine Learning–ICANN*, 2011, pp. 10–17.

[22] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015, pp. 1–15.