Stochastic Convolutional Recurrent Networks

Jen-Tzung Chien
Department of Electrical and Computer Engineering
National Chiao Tung University
Hsinchu, Taiwan

Yu-Min Huang
Department of Electrical and Computer Engineering
National Chiao Tung University
Hsinchu, Taiwan

***Abstract*—Recurrent neural network (RNN) has been widely used for sequential learning which has achieved a great success in different tasks. The temporal convolutional network (TCN), a variant of one-dimensional convolutional neural network (CNN), was also developed for sequential learning in presence of sequence data. RNN and TCN typically captures long-term and short-term features in temporal or spatial domain, respectively. This paper presents a new sequential learning, called the convolutional recurrent network (CRN), which fulfills TCN as an encoder and RNN as a decoder so that the *global semantics* as well as the *local dependencies* are simultaneously characterized from sequence data. To facilitate the interpretation and robustness in neural models, we further develop the stochastic modeling for CRN based on variational inference. The merits of CNN and RNN are then incorporated in inference of latent space which sufficiently produces a generative model for sequential prediction. Experiments on language model shows the effectiveness of stochastic CRN when compared with the other sequential machines.**

***Index Terms*—Convolutional neural network, recurrent neural network, stochastic modeling, sequential learning**

I. Introduction

Deep learning has been successfully developed for numerous applications in signal processing, natural language processing [1]–[5] and computer vision [6], [7]. Basically, deep models can handle high-dimensional data with the complicated mapping between input signals and output targets, and perform well for different classification and regression tasks. Nevertheless, it is still challenging to carry out a desirable generation task in presence of high-dimensional data. Meanwhile, sequence data in temporal and spatial domains are everywhere in real world and is ranged from speech signals to music signals, natural sentences and video streams, to name a few. When we deal with sequential learning and generation, it is important to predict or generate future targets based on all previous samples due to the casual property in signals. Such a prediction is called the autoregressive generation where the prediction at each time step is conditioned on all previous observations. Autoregressive generation is seen as a building block in many systems with temporal and spatial signals. This paper presents a new stochastic sequential learning [8]–[11] for autoregressive generation where an inference and generative procedure based on convolutional neural network (CNN) [12]–[16] and recurrent neural network (RNN) is developed.

RNN [17]–[19] is specialized as a recurrent machine which identifies the temporal or spatial features from sequence patterns. The dynamic state or internal memory is evolved through time. RNN has been recognized as a popular solution to autoregressive model in different practical systems. Recently, the temporal convolutional network (TCN) [20], [21] was proposed for sequential learning using temporal data in spite of many successful spatial models for image data using CNN in computer vision. Typically, TCN is beneficial for parallel computation which provides rapid prediction. Multi-layer TCN can capture the temporal hierarchy where different layers represent various sizes of receptive field. RNN and TCN are both feasible to sequential modeling. This study aims to combine TCN and RNN in construction of the so-called convolutional recurrent network (CRN) for sequential learning where long- and short-term temporal patterns are learned [22].

Basically, TCN is powerful to learn from sequence data to extract short-term features in local fields while RNN is specialized to capture long-term semantics in global contexts. The proposed CRN would like to infer or encode local information via convolutional layers and then generate or decode each individual time sample via recurrent layers. CRN corresponds to implement TCN as encoder and RNN as decoder. A hybrid model of TCN and RNN is established. The complementary local and global features are characterized. Importantly, the recurrent layers in CRN are used to relax the limitation of TCN where the size of receptive field is constrained by the number of layers. CRN allocates the recurrent layers on top of convolutional layers so that the insufficiency of long-term temporal characteristics in TCN can be compensated. Furthermore, the stochastic variant of CRN (SCRN) is proposed to improve the robustness of CRN for sequential prediction. The randomness of sequential latent variables is reflected in optimization procedure via variational inference [9], [23] where the variational lower bound of log likelihood, marginalized over latent variables, is maximized. SCRN is proposed with an explainable latent space. The experiments on language modeling are conducted to investigate the performance of different convolutional recurrent networks. We show the merits of the proposed methods by comparing with RNN and TCN under different experimental settings.

II. Background and Motivation

This paper presents a new neural network architecture which combines CNN and RNN for sequential learning. In the literature, it has been common to mix CNN and RNN for speech, image and video processing in different tasks and applications. For example, two-dimensional (2-D) CNN was

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

used for representation of images while RNN was applied to capture temporal relations in video data [24]. In [25]–[27], CNN was concatenated with RNN for object recognition, sign language recognition and person identification. The hybrid CNN and RNN was also exploited for natural language processing in [28], [29]. In previous hybrid CNN and RNN models, the spatial model using 2-D CNN was employed. These models are different from 1-D CNN or TCN where the causality and dilation [30] are considered. To facilitate sequential learning, TCN can act as a meaningful approach to capture temporal hierarchy. The upper layers are feasible to span larger receptive field with increasing window size. This paper constructs the temporal hierarchy by using TCN where RNN is further allocated in top layer. The receptive field with infinitely large size is represented in the proposed CRN.

Besides, we explore the stochastic latent space for CRN by using variational inference. Previously, stochastic modeling has improved the generalization in different learning tasks. In [31], a stochastic variant of RNN was proposed to build a recurrent latent variable model. Recently, the stochastic property was merged in TCN [32], [33] where RNN was excluded in temporal modeling. In [31]–[33], an additional latent variable was added to fulfill stochastic modeling in sequential learning based on variational inference. A simple linear transformation was used in either encoder or decoder. This study presents how a variational or stochastic CRN is formulated to improve sequential learning where TCN and RNN separately act as the encoder for local features and decoder for global views in inference and generation stages, respectively. A meaningful two-stage inference is designed similar to variational autoencoder (VAE) [23].

III. Convolutional Recurrent Network

Temporal convolutional network (TCN) leverages large receptive field by stacking a number of dilated convolutions. The size of receptive fields is determined by the number of layers. In general, TCN introduces a temporal hierarchy where the upper layers can access longer sub-sequences of input signals so as to learn representations at a larger time scale. TCN has an attractive architecture to capture temporal dependency at various time scales. On the other hand, RNN with gating mechanism [18], [34], [35] is feasible to capture temporal dependency with unbounded length. It is therefore meaningful to introduce RNN as the most upper layer so as to relax the limitation in TCN. Here comes the deterministic variant of convolutional recurrent network (CRN).

The architecture of CRN is shown in Figure 1 with input layer $\{\mathbf{x}_t\}$ and output layer $\{\mathbf{y}_t\}$ where RNN is built on top of TCN. TCN is seen as 1-D CNN with kernel size 2 and two hidden layers. Dilation is 1 and 2 in first layer $\{\mathbf{d}_t\}$ and second hidden layer $\{\mathbf{z}_t\}$, respectively. Information propagation is run in a bottom-up manner. More hidden layers can be applied. TCN acts as an encoder to extract local temporal features $\{\mathbf{z}_t\}$ with a receptive field containing four time steps while RNN serves as a decoder to capture global view via long-term recurrent codes $\{\mathbf{h}_t\}$. The size of receptive field

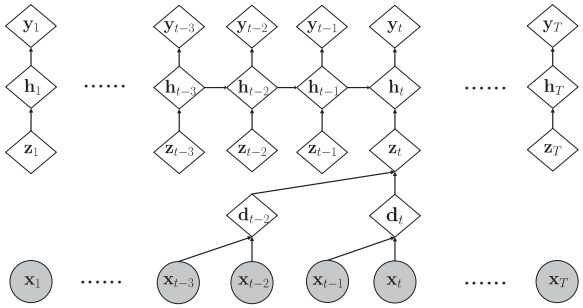

Fig. 1. Architecture of convolutional recurrent network.

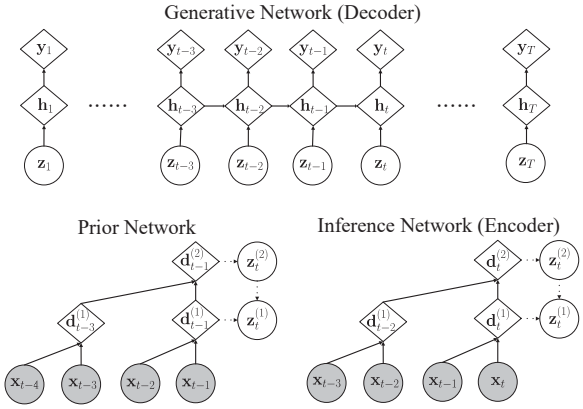

Fig. 2. Architecture of stochastic convolutional recurrent network.

is unbounded in RNN. Basically, convolution operation can extract local features efficiently as shown in many computer vision tasks. Nevertheless, in sequential learning, TCN still suffers from the limited window of receptive field caused by the specifications of dilation and kernel size. To relax this limitation, RNN is combined as a decoder to continuously capture long-term temporal semantics from the beginning $\mathbf{x}_1$. TCN and RNN are complementary with different functions which are mutually leveraged to build CRN. Take text modeling as an example, TCN encoder can be regarded as a new type of *word embedding*. Instead of embedding every words independently, TCN encodes a word by relating to its neighboring words. This is reasonable because there might be different meanings for a single word. Contextual information provides an efficient and meaningful way to calculate the *causal* and *dilated convolutional embedding* $\mathbf{z}_t$ for each word $\mathbf{x}_t$. Taking a look at previous words $\{\mathbf{x}_i\}_{i=t-3}^{t}$ is helpful to find precise meaning or embedding of current word $\mathbf{z}_t$. RNN decoder can accordingly identify long-range information from full text paragraph with limited computation overhead. This RNN extends the temporal hierarchy since the size of receptive field is continuously increasing to final time step T .

IV. Stochastic Neural Network

The combination of TCN and RNN can improve model capability in the resulting CRN. However, the enhanced model

usually induces the issue of overfitting. Pursuing stochastic property for an existing deterministic model is beneficial to handle this issue. Here, variational inference provides a theoretical bound for generation. Moreover, stochastic variants of autoregressive models in previous works [31]–[33], [36], [37] have shown significant improvement in characterizing complex and structural relation between $\{\mathbf{x}_t\}$ and $\{\mathbf{y}_t\}$. This study is accordingly motivated by implementing the stochastic CRN (SCRN) where the word embeddings from TCN encoder $\{\mathbf{z}_t\}$ are *random* as illustrated by circles in Figure 2. SCRN differs from CRN where *deterministic* variables $\{\mathbf{z}_t\}$ are assumed as shown by diamonds in Figure 1. SCRN is constructed with an inference model as encoder and a generative model as decoder. TCN is applied to infer or encode a Gaussian variable $\mathbf{z}_t$ from input signal $\mathbf{x}_t$ with mean $\boldsymbol{\mu}_t$ and standard deviation $\boldsymbol{\sigma}_t$ while RNN is adopted to generate output signal $\mathbf{y}_t$ from a set of Gaussian samples $\mathbf{z}_t$. The whole procedure of inference and generation is governed by variational inference as detailed as follows.

First, the deterministic hidden state $\mathbf{d}_t^{(l)}$ at layer l and time t is calculated by 1-D convolution in a bottom-up manner by

$$\mathbf{d}_t^{(l)} = \text{Conv}\left(\mathbf{d}_t^{(l-1)}, \mathbf{d}_{t-j}^{(l-1)}\right), \quad 1 \le l \le L \tag{1}$$

where $\mathbf{d}_t^{(0)} \triangleq \mathbf{x}_t$ and $j = 2^{l-1}$ means the dilation. This hidden state $\mathbf{d}_t^{(l)}$ summarizes previous input signals $\mathbf{x}_{\le t} = \{\mathbf{x}_i\}_{i \le t}$ within a receptive field ended at time t. The latent codes $\{\mathbf{d}_t^{(l)}\}$ are then transformed by a linear matrix to calculate Gaussian mean $\boldsymbol{\mu}_t$ and standard deviation $\boldsymbol{\sigma}_t$ for sampling of the latent variables $\{\mathbf{z}_t^{(l)}\}$ (shown by dashed lines). Importantly, each sample $\mathbf{z}_t^{(l)}$ at layer l and time t is conditioned on the latent code $\mathbf{z}_t^{(l+1)}$ at a higher layer $l+1$ at time t and the deterministic state $\mathbf{d}_{t-1}^{(l)}$ at layer l at previous time $t-1$ [38]. Using SCRN, the inference model is built according to a *prior network* of $\mathbf{z}_t$ given by history samples $\mathbf{x}_{<t}$

$$p_\omega(\mathbf{z}_t|\mathbf{x}_{<t}) = p_\omega(\mathbf{z}_t^{(L)}|\mathbf{d}_{t-1}^{(L)}) \prod_{l=1}^{L-1} p_\omega(\mathbf{z}_t^{(l)}|\mathbf{z}_t^{(l+1)}, \mathbf{d}_{t-1}^{(l)}) \tag{2}$$

where

$$p_\omega(\mathbf{z}_t^{(l)}|\mathbf{z}_t^{(l+1)}, \mathbf{d}_{t-1}^{(l)}) = \mathcal{N}(\boldsymbol{\mu}_{p,t}^{(l)}, \boldsymbol{\sigma}_{p,t}^{(l)}) \tag{3}$$

with Gaussian parameters

$$[\boldsymbol{\mu}_{p,t}^{(l)}, \boldsymbol{\sigma}_{p,t}^{(l)}] = f_p^{(l)}(\mathbf{z}_t^{(l+1)}, \mathbf{d}_{t-1}^{(l)}) \tag{4}$$

calculated by a fully connected (FC) network $f_p^{(l)}(\cdot)$ with parameter ω using $\mathbf{d}_{t-1}^{(l)}$ at time $t-1$. Notably, latent code $\mathbf{z}$ is computed in a top-down order, which is different from bottom-up order for $\mathbf{d}$. The underlying reason is to pursue a latent code $\mathbf{z}$ with rough global feature in higher layer in the beginning and then with delicate local feature in lower layer in learning procedure.

Variational inference [39], [40] is introduced to infer a variational posterior of $\mathbf{z}_t$ for prediction by using not only history samples $\mathbf{x}_{<t} = \{\mathbf{x}_i\}_{i=1}^{t-1}$ but also current sample $\mathbf{x}_t$, namely $\mathbf{x}_{\le t}$

$$q_\phi(\mathbf{z}_t|\mathbf{x}_{\le t}) = q_\phi(\mathbf{z}_t^{(L)}|\mathbf{d}_t^{(L)}) \prod_{l=1}^{L-1} q_\phi(\mathbf{z}_t^{(l)}|\mathbf{z}_t^{(l+1)}, \mathbf{d}_t^{(l)}) \tag{5}$$

where

$$q_\phi(\mathbf{z}_t^{(l)}|\mathbf{z}_t^{(l+1)}, \mathbf{d}_t^{(l)}) = \mathcal{N}(\boldsymbol{\mu}_{q,t}^{(l)}, \boldsymbol{\sigma}_{q,t}^{(l)}) \tag{6}$$

with Gaussian parameters

$$[\boldsymbol{\mu}_{q,t}^{(l)}, \boldsymbol{\sigma}_{q,t}^{(l)}] = f_q^{(l)}(\mathbf{z}_t^{(l+1)}, \mathbf{d}_t^{(l)}). \tag{7}$$

Here, $f_q^{(l)}(\cdot)$ denotes a variational FC network where the input $\mathbf{d}_t^{(l)}$ at time t is used. Notably, the history posterior in Eq. (2) using previous sample $\mathbf{x}_{<t}$ is treated as the *prior* for an *inference network* as variational posterior at current time with sample $\mathbf{x}_t$ in Eq. (5). Variational parameter ϕ contains those from 1-D convolution and variational FC network. Both parameters ω and ϕ are varied at each layer l. Overall, the evidence lower bound (ELBO) $\mathcal{L}$ of log conditional likelihood $\log p(\mathbf{y}|\mathbf{x})$ is formulated as

$$\begin{aligned}\log p(\mathbf{y}|\mathbf{x}) \ge & \mathbb{E}_{q_\phi(\mathbf{z} \le T|\mathbf{x} \le T)}\left[\sum_{t=1}^{T} \log p_\theta(\mathbf{y}_t|\mathbf{x}_{\le t}, \mathbf{z}_{\le t})\right. \\ & \left. - \mathcal{D}_{\text{KL}}(q_\phi(\mathbf{z}_t|\mathbf{x}_{\le t}) \| p_\omega(\mathbf{z}_t|\mathbf{x}_{<t}))\right] \triangleq \mathcal{L}\end{aligned} \tag{8}$$

where $\mathcal{D}_{\text{KL}}(\cdot)$ denotes the Kullback-Leibler divergence. Here, the *generative network* for output sample $\mathbf{y}_t$ at each time t is calculated by using an RNN (or a long short-term memory (LSTM)) $f_\theta(\cdot)$ given by a hidden state $\mathbf{h}_t$ and a concatenated input vector

$$\mathbf{z}_t = [(\mathbf{z}_t^{(1)})^\top \cdots (\mathbf{z}_t^{(L)})^\top]^\top$$

from the samples in different layers in inference distribution $p_\phi(\mathbf{z}_t|\mathbf{x}_{\le t})$. The generative distribution $p_\theta(\mathbf{y}_t|\mathbf{x}_{\le t}, \mathbf{z}_{\le t})$ in decoder is driven by LSTM hidden state $\mathbf{h}_t$ and calculated by

$$p_\theta(\mathbf{y}_t|\mathbf{h}_t) = \text{Mult}(f_\theta), \quad \text{where} \quad \mathbf{h}_t = f_\theta(\mathbf{z}_t, \mathbf{h}_{t-1}) \tag{9}$$

where a multinomial probability is continuously updated for predicting each output word or target $\mathbf{y}_t$ at time t based on a hidden state $\mathbf{h}_t$ updated by $f_\theta(\cdot)$ using the input from stochastic state of TCN encoder $\mathbf{z}_t$. LSTM decoder contains parameter θ. A stochastic variant of hybrid TCN and LSTM is implemented according to the stochastic gradient descent algorithm using the gradients $\{\frac{\partial \mathcal{L}}{\partial \omega}, \frac{\partial \mathcal{L}}{\partial \phi}, \frac{\partial \mathcal{L}}{\partial \theta}\}$ for updating prior network, inference network and generative network, respectively.

V. Experiments

Penn Treebank (PTB) dataset [41] was used to evaluate word prediction in sequential learning for language modeling [42]–[45]. PTB contained 929K training words, 73K validation words, 82K test words and 10K words in its dictionary. This dataset was preprocessed by removing numbers and punctuations and lower-casing the capital letters. Perplexity is measured to illustrate how well a probability distribution

or model predicts a future word. Lower perplexity generally implies that better performance is achieved for word prediction. LSTM (here denoted as RNN), TCN [20], [21], stochastic TCN (STCN) [33], CRN and stochastic CRN (SCRN) were implemented. Different models were trained by running twenty epochs using stochastic gradient descent algorithm [46], [47]. The mini-batch size was twenty. Gradient clipping was applied to avoid gradient vanishing [48]. For consistency, all parameters were uniformly initialized between -1 and 1. The size of hidden states and the amount of kernels were 450 for all models using LSTM. Model size was also included in the evaluation.

TABLE I
PERPLEXITY AND MODEL SIZE USING DIFFERENT COMBINATIONS OF TCN AND RNN.

Model	Size	Train	Validation	Test
RCN	9.5M	125	129	125
2-layer RNN	10.4M	90	126	122
CRN	11.0M	84	128	123

We claim that it is beneficial to encode local information using TCN earlier than long term information using RNN. It is important to evaluate different two-stage architectures under comparable model size as shown in Table I. The recurrent convolutional network (RCN) is implemented as RNN encoder and TCN decoder. 2-layer RNN means RNN with two layers. CRN obtains the lowest perplexity in training phase but comparable perplexity in test phase. Overfitting issue happens in CRN. The proposed SCRN may deal with this issue. Figures 3 and 4 illustrate the learning curves of perplexity using training and test data, respectively. Typically, CRN and SCRN significantly perform better than the other models in training phase. RNN learns very quick and converges very soon due to the limited model capacity. TCN and STCN perform worse than RNN because they cannot capture very long term information. CRN combine the advantages of RNN and TCN, so its modeling ability is better than both of them. CRN suffers from overfitting issue. With the stochastic property, SCRN is more robust and more accurate in word prediction than CRN during test phase. Table II compares different models in terms of model size and perplexities of training, validation and test data. CRN captures the temporal dependencies better than RNN (or LSTM) and TCN. STCN and SCRN perform better than TCN and CRN in perplexity, respectively. But, the model size is increased as well due to additional memory cost from prior network and inference network in STCN and SCRN. With the stochastic property, SCRN outperforms the other models in both training and test phases.

VI. CONCLUSIONS

We have presented a new two-stage neural network model for sequential learning. This model combined the advantages of temporal convolutional network and recurrent neural network to capture complementary temporal features to characterize long-term semantics and short-term dependencies in natural language, respectively. Importantly, stochastic modeling in convolutional recurrent network was proposed to improve the robustness and expressiveness in sequential machine where the temporal hierarchy in an extended receptive field was learned. Experiments on language model illustrated the merit of the proposed method. Future works include the extension to other sequential learning tasks. Multi-scale temporal dependency will be explored and combined with individual recurrent nets.

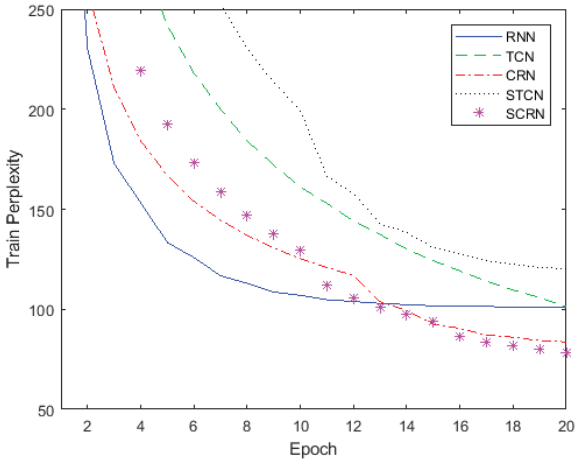

Fig. 3. Perplexity of different models using training data.

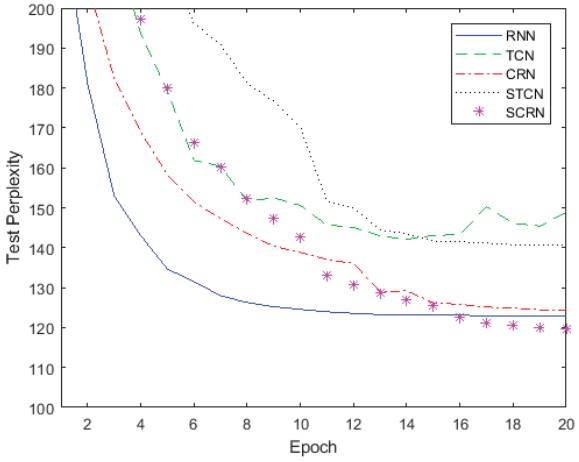

Fig. 4. Perplexity of different models using test data.

TABLE II
PERPLEXITY AND MODEL SIZE USING DIFFERENT MODELS.

Model	Size	Train	Validation	Test
RNN	8.8M	101	128	123
TCN	5.2M	100	152	146
STCN	23.4M	119	147	140
CRN	11.0M	84	128	123
SCRN	17.6M	78	125	119

Attention mechanism will be developed.

REFERENCES

[1] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2015, pp. 4580–4584.

[2] J. Gehring, M. Auli, D. Grangier, and Y. Dauphin, "A convolutional encoder model for neural machine translation," in *Proc. of Annual Meeting of Association for Computational Linguistics*, 2017, vol. 1, pp. 123–135.

[3] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proc. of International Conference on Machine Learning*, 2017, pp. 933–941.

[4] J.-T. Chien and C.-H. Lee, "Deep unfolding for topic models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 2, pp. 318–331, 2017.

[5] Y. Tu, M.-W. Mak, and J.-T. Chien, "Information maximized variational domain adversarial learning for speaker verification," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 6449–6453.

[6] A. Van Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proc. of International Conference on Machine Learning*, 2016, pp. 1747–1756.

[7] J.-T. Chien and Y.-T. Bao, "Tensor-factorized neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 5, pp. 1998–2011, 2018.

[8] J.-T. Chien, "Deep Bayesian natural language processing," in *Proc. of Annual Meeting of Association for Computational Linguistics : Tutorial Abstracts*, 2019.

[9] J.-T. Chien and C. Shen, "Stochastic recurrent neural network for speech recognition," in *Proc. of Annual Conference of International Speech Communication Association*, 2017, pp. 1313–1317.

[10] J.-T. Chien and K.-T. Kuo, "Variational recurrent neural networks for speech separation," in *Proc. of Annual Conference of International Speech Communication Association*, 2017, pp. 1193–1197.

[11] J.-T. Chien, "Deep Bayesian mining, learning and understanding," in *Proc. of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, p. 3197–3198.

[12] Y. LeCun, Y. Bengio, et al., "Convolutional networks for images, speech, and time series," *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10, pp. 1995, 1995.

[13] Z. Yang, Z. Hu, R. Salakhutdinov, and T. Berg-Kirkpatrick, "Improved variational autoencoders for text modeling using dilated convolutions," in *Proc. of International Conference on Machine Learning*, 2017, pp. 3881–3890.

[14] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. of International Conference on Learning Representation*, 2016.

[15] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in Neural Information Processing Systems*, pp. 649–657. 2015.

[16] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *Proc. of International Conference on Machine Learning*, 2017, pp. 1243–1252.

[17] J. L. Elman, "Finding structure in time," *Cognitive Science*, vol. 14, no. 2, pp. 179–211, 1990.

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

[19] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *The Journal of Machine Learning Research*, vol. 3, pp. 115–143, 2003.

[20] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[21] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[22] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *Proc. of International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 2018, pp. 95–104.

[23] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. of International Conference on Learning Representations*, 2014.

[24] Y.-M. Huang, H.-H. Tseng, and J.-T. Chien, "Stochastic fusion for multi-stream neural network in video classification," in *Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 2019, pp. 69–74.

[25] M. Liang and X. Hu, "Recurrent convolutional neural network for object recognition," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3367–3375.

[26] R. Cui, H. Liu, and C. Zhang, "Recurrent convolutional neural networks for continuous sign language recognition by staged optimization," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7361–7369.

[27] N. McLaughlin, J. Martinez del Rincon, and P. Miller, "Recurrent convolutional network for video-based person re-identification," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1325–1334.

[28] X. Wang, W. Jiang, and Z. Luo, "Combination of convolutional and recurrent neural network for sentiment analysis of short texts," in *Proc. of International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 2428–2437.

[29] X. Zhang, F. Chen, and R. Huang, "A combination of RNN and CNN for attention-based relation classification," *Procedia Computer Science*, vol. 131, pp. 911–917, 2018.

[30] S. Chang, Y. Zhang, W. Han, M. Yu, X. Guo, W. Tan, X. Cui, M. Witbrock, M. A. Hasegawa-Johnson, and T. S. Huang, "Dilated recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 77–87.

[31] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Advances in Neural Information Processing Systems*, 2015, pp. 2980–2988.

[32] G. Lai, B. Li, G. Zheng, and Y. Yang, "Stochastic wavenet: A generative latent variable model for sequential data," *arXiv preprint arXiv:1806.06116*, 2018.

[33] E. Aksan and O. Hilliges, "STCN: Stochastic temporal convolutional networks," *arXiv preprint arXiv:1902.06568*, 2019.

[34] K. Cho, Bart Van M., C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[35] C. Tallec and Y. Ollivier, "Can recurrent neural networks warp time?," *arXiv preprint arXiv:1804.11188*, 2018.

[36] J.-T. Chien and C.-W. Wang, "Variational and hierarchical recurrent autoencoder," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 3202–3206.

[37] S. Watanabe and J.-T. Chien, *Bayesian Speech and Language Processing*, Cambridge University Press, 2015.

[38] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther, "Ladder variational autoencoders," in *Advances in Neural Information Processing Systems*, 2016, pp. 3738–3746.

[39] M. J. Beal, *Variational Algorithms for Approximate Bayesian Inference*, Ph.D. thesis, University of London, 2003.

[40] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. of International Conference on Machine Learning*, 2014, pp. 1278–1286.

[41] T. Mikolov and G. Zweig, "Context dependent recurrent neural network language model," in *Proc. of IEEE Workshop on Spoken Language Technology*, 2012, pp. 234–239.

[42] J.-T. Chien, "Association pattern language modeling," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1719–1728, 2006.

[43] J.-T. Chien, "Hierarchical Pitman-Yor-Dirichlet language model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 8, pp. 1259–1272, 2015.

[44] J.-T. Chien and Y.-C. Ku, "Bayesian recurrent neural network for language modeling," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 361–374, 2016.

[45] J.-T. Chien and C.-Y. Kuo, "Markov recurrent neural network language model," in *Proc. of IEEE Automatic Speech Recognition and Understanding Workshop*, 2019, pp. 807–813.

[46] H. Robbins and S. Monro, "A stochastic approximation method," *The Annals of Mathematical Statistics*, pp. 400–407, 1951.

[47] J. Kiefer, J. Wolfowitz, et al., "Stochastic estimation of the maximum of a regression function," *The Annals of Mathematical Statistics*, vol. 23, no. 3, pp. 462–466, 1952.
[48] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.