INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Semi-supervised Prosody Modeling Using Deep Gaussian Process Latent Variable Model

Tomoki Koriyama[1†], *Takao Kobayashi*[1]

[1]Tokyo Institute of Technology, Japan

t.koriyama@ieee.org, takao.kobayashi@ip.titech.ac.jp

Abstract

This paper proposes a semi-supervised speech synthesis framework in which prosodic labels of training data are partially annotated. When we construct a text-to-speech (TTS) system, it is crucial to use appropriately annotated prosodic labels. For this purpose, manually annotated ones would provide a good result, but it generally costs much time and patience. Although recent studies report that end-to-end TTS framework can generate natural-sounding prosody without using prosodic labels, this does not always appear in arbitrary languages such as pitch accent ones. Alternatively, we propose an approach to utilizing a latent variable representation of prosodic information. In the latent variable representation, we employ deep Gaussian process (DGP), a deep Bayesian generative model. In the proposed semi-supervised learning framework, the posterior distributions of latent variables are inferred from linguistic and acoustic features, and the inferred latent variables are utilized to train a DGP-based regression model of acoustic features. Experimental results show that the proposed framework can give a comparable performance with the case using fully-annotated speech data in subjective evaluation even if the prosodic information of pitch accent is limited.

Index Terms: deep Gaussian process, statistical speech synthesis, latent variable model, prosody, semi-supervised learning

1. Introduction

Prosody is an important factor affecting the perceptual quality of synthetic speech in text-to-speech (TTS) systems. In statistical speech synthesis frameworks, prosodic features, such as fundamental frequency (F0), duration, and power, are modeled using speech recordings and their corresponding linguistic information called contexts. To generate natural-sounding prosody, we should make prosodic context labels that describe the recorded speech utterances accurately. However, it is not always easy to make such context labels from transcription texts because prosodic information varies much depending on situations and speakers. To avoid the mismatch of acoustic features and context labels of training data, we generally need to annotate the data manually. However, this leads to the cost of much time and patience. Moreover, inconsistencies between prosodic labels annotated by different annotators often occur because prosodic labels are influenced by annotators' perception.

One of the solutions to overcome these problems is to employ automatic prosodic labeling based on statistical models such as hidden Markov model (HMM) [1], conditional random field (CRF) [2], CRF-HMM hybrid model [3], and recurrent neural network (RNN) [4]. These methods directly predict the prosodic events using another database whose prosodic labels are already obtained. Although these methods perform well if we can prepare a large amount of annotated data, the collection of such data also requires much time and cost.

Another approach is to use a state-of-the-art end-to-end framework, in which the relationships between character sequences and acoustic feature sequences are directly modeled without using prosodic contexts [5, 6]. It has been reported that the characteristics of prosody in English, such as stress, are automatically modeled by using a neural-network-based sequence-to-sequence model [5]. However, in Japanese, which is a pitch accent language and accent information is not described in texts, it has been reported that the naturalness of synthetic speech degrades when we do not use accent information as an input [7].

In this paper, we focus on the framework using latent variables as the alternatives to prosodic labels proposed by Moungsri et al. [8]. This framework uses Gaussian process latent variable model (GPLVM) [9], which is a Bayesian generative model, to represent pitch contours of a tonal language by a low-dimensional latent space. By using the latent variable representation as context, we can augment the training data and achieve semi-supervised or unsupervised learning [10]. However, the limitation of GPLVM was also reported in [8], that is, single latent space is insufficient for prosodic labeling. This is because a single-layer GPLVM has low expressiveness for complicated features such as pitch contours. Moreover, this GPLVM-based framework is hard to introduce linguistic features despite the fact that prosody is generally affected by the contents of text input.

To achieve semi-supervised prosody modeling, in this study, we propose a latent variable representation framework using deep Gaussian processes (DGPs) [11]. The basic idea is based on the speech synthesis framework using DGP regression [12]. In the proposed framework, we consider a generative model in which acoustic features including pitch and spectral information are generated from the linguistic information unrelated to prosody and the latent variables of phrase-level prosodic information. By using the variational Bayesian framework for DGP-based latent variable model, the posteriors of latent variables are inferred from not only acoustic features but also linguistic information. We can perform semi-supervised learning using the predicted latent variables in the same way as the single-layer GPLVM.

In the experimental evaluations, we use Japanese database whose accent type information is partially observed in the training data set. We represent unobserved accent type information using mora- and accent-phrase-level latent spaces. We evaluate the performance of the proposed semi-supervised prosody modeling method objectively and subjectively, and show that the proposed method can give comparable naturalness with the case using fully-annotated speech data in subjective evaluation even if the prosodic information of pitch accent is limited.

[†]Currently, the author is with the University of Tokyo.

http://dx.doi.org/10.21437/Interspeech.2019-2497

2. Background of GP and DGP

We first describe an overview of Gaussian processes (GPs) and DGPs. In machine learning using GPs [13], it is assumed that a function, which represents the relationship between input $\mathbf{x}$ and output y, is a sample of GP. This is expressed by the following equations:

$$y = f(\mathbf{x}) + \epsilon \tag{1}$$

$$f \sim \mathcal{GP}(m(\cdot), k(\cdot, \cdot)) \tag{2}$$

where ϵ is a random noise and $m(\cdot)$ and $k(\cdot, \cdot)$ are mean and kernel functions, respectively. In GP-based regression, we infer the predictive probability for new input, $p(y^*|\mathbf{x}^*, \mathbf{Y}, \mathbf{X})$, using the data of input and output pairs $(\mathbf{X}, \mathbf{Y})$.

In contrast to the GP regression, Bayesian Gaussian process latent variable model (GPLVM) [9] infer the posterior probability of inputs $\mathbf{X}$ under the condition where only outputs $\mathbf{Y}$ are given. The Bayesian GPLVM is a non-linear and Bayesian generative model and can be used for unsupervised learning such as dimension reduction. Damianou and Lawrence [10] applied the combination of GP regression and GPLVM to semi-supervised learning for two cases: input features are missing[1] or output features are missing.

Although the GP regression and GPLVM are effective in various problems, single-layer GPs have limitations in performance caused by their kernel functions, which should be chosen carefully. To overcome the limitation of kernel functions, Damianou and Lawrence [11] proposed deep Gaussian processes. In the deep Gaussian processes, it is assumed that the latent function f is decomposed into multiple functions, and each function is a sample of a Gaussian process, which is given by the following equations:

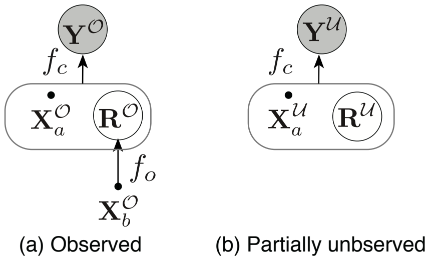
$$f = f^L \circ f^{L-1} \circ \cdots \circ f^1 \tag{3}$$

$$f^\ell \sim \mathcal{GP}(m^\ell(\cdot), k^\ell(\cdot, \cdot)) \tag{4}$$

where L is the number of layers. Against the problem that the computation for DGP is intractable, Salimbeni and Deisenroth [14] proposed an approximation method referred to as doubly stochastic variational inference (DSVI). In DSVI-based DGP, hyperparameters and variational parameters are optimized by maximizing the following evidence lower bound (ELBO):

$$\mathcal{L} = \frac{1}{S}\sum_{s=1}^{S}\sum_{i=1}^{N}\Bigg\{\sum_{d=1}^{D}\mathbb{E}_{q(\mathrm{f}_{i,s}^d|\hat{\mathbf{h}}_{i,s}^{L-1})}\left[\log p\left(y_i^d|\mathrm{f}_{i,s}^d\right)\right] - \frac{S}{N}\sum_{\ell=1}^{L}\mathrm{KL}(q(\mathbf{U}^\ell)\|p(\mathbf{U}^\ell|\mathbf{Z}^\ell))\Bigg\} \tag{5}$$

where N and D are the numbers of training data points and the dimensionality of output variable, respectively. S is the number of Monte Carlo sampling to obtain hidden layer sample $\hat{\mathbf{h}}_{is}^{L-1}$ of the $(L-1)$-th layer. y_i^d and $\mathrm{f}_{i,s}^d$ correspond to the observed and function output variables of dimension d and sample index s. $\mathbf{Z}^\ell$ and $\mathbf{U}^\ell$ are inducing inputs and outputs, which mimic representative points of training data. KL denotes Kullback-Leibler divergence. The parameter training can be performed efficiently by gradient-based stochastic optimization because the ELBO in (5) is calculated by the sum of respective data points.



$\mathbf{Y}^{\mathcal{O}}$
f_c
$\mathbf{X}_a^{\mathcal{O}}$
$\mathbf{R}^{\mathcal{O}}$
f_o
$\mathbf{X}_b^{\mathcal{O}}$
(a) Observed
$\mathbf{Y}^{\mathcal{U}}$
f_c
$\mathbf{X}_a^{\mathcal{U}}$
$\mathbf{R}^{\mathcal{U}}$
(b) Partially unbserved

Figure 1: *DGP model for semi-supervised learning using observed and partially unobserved data.*

3. Semi-supervised learning using DGP

In this study, we attempt to apply the semi-supervised training method proposed in [11] to the framework of DSVI-based DGP. We assume the case that training data includes data points whose input features are partially unobserved. We represent the unobserved data as $\mathcal{D}^{\mathcal{U}} = (\mathbf{X}_a^{\mathcal{U}}, \mathbf{Y}^{\mathcal{U}})$ and let $\mathcal{D}^{\mathcal{O}} = (\mathbf{X}_a^{\mathcal{O}}, \mathbf{X}_b^{\mathcal{O}}, \mathbf{Y}^{\mathcal{O}})$ be fully observed data, which does not have missing information. The subscripts a and b denote that a is observed in the whole data and b is missing in unobserved data. Figure 1 shows the graphical representation of probabilistic model proposed in this study. For unobserved data, we assume that the output $\mathbf{Y}$ is generated from input feature $\mathbf{X}_a$ and latent variable $\mathbf{R}$ using a latent function f_c. For observed data, in contrast, $\mathbf{R}$ is predicted from input feature $\mathbf{X}_b$ using a latent function f_o, and $\mathbf{Y}$ is generated in the same way as unobserved data.

The marginal likelihood of whole of training data is given by

$$\begin{aligned} &p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{U}}|\mathbf{X}_a^{\mathcal{O}}, \mathbf{X}_a^{\mathcal{U}}, \mathbf{X}_b^{\mathcal{O}}) \\ &= \int p(\mathbf{Y}^{\mathcal{O}}, \mathbf{Y}^{\mathcal{U}}|\mathbf{R}^{\mathcal{O}}, \mathbf{R}^{\mathcal{U}}, \mathbf{X}_a^{\mathcal{O}}, \mathbf{X}_a^{\mathcal{U}}) \\ &\qquad p(\mathbf{R}^{\mathcal{O}}|\mathbf{X}_b^{\mathcal{O}})p(\mathbf{R}^{\mathcal{U}})d\mathbf{R}^{\mathcal{O}}d\mathbf{R}^{\mathcal{U}}. \end{aligned} \tag{6}$$

Using DSVI [14], the variational lower bound of log marginal likelihood is given by

$$\mathcal{L} = \mathcal{L}^{\mathcal{O}} + \mathcal{L}^{\mathcal{U}} \tag{7}$$

$$\begin{aligned} \mathcal{L}^{\mathcal{O}} &= \frac{1}{S}\sum_{s=1}^{S}\sum_{i\in\mathcal{I}^O}\sum_{d=1}^{D}\mathbb{E}_{q(\mathrm{f}_{i,s}^d|\hat{\mathbf{h}}_{i,s}^{L-1})}\left[\log p(y_i^d|\mathrm{f}_{i,s}^d)\right] \\ &\quad - \sum_{\ell\in L(f_c)}\mathrm{KL}(q(\mathbf{U}^\ell)\|p(\mathbf{U}^\ell|\mathbf{Z}^\ell)) \\ &\quad - \sum_{\ell\in L(f_o)}\mathrm{KL}(q(\mathbf{U}^\ell)\|p(\mathbf{U}^\ell|\mathbf{Z}^\ell)) \end{aligned} \tag{8}$$

$$\begin{aligned} \mathcal{L}^{\mathcal{U}} &= \frac{1}{S}\sum_{s=1}^{S}\sum_{i\in\mathcal{I}^U}\Big\{\sum_{d=1}^{D}\mathbb{E}_{q(\mathrm{f}_{i,s}^d|\hat{\mathbf{h}}_{is}^{L-1})}\left[\log p(y_i^d|\mathrm{f}_{i,s}^d)\right] \\ &\qquad - S\cdot\mathrm{KL}(q(\mathbf{r}_i)||p(\mathbf{r}_i))\Big\} \\ &\quad - \sum_{\ell\in L(f_c)}\mathrm{KL}(q(\mathbf{U}^\ell)\|p(\mathbf{U}^\ell|\mathbf{Z}^\ell)). \end{aligned} \tag{9}$$

The terms $\mathcal{L}^{\mathcal{O}}$ and $\mathcal{L}^{\mathcal{U}}$ are the partial lower bounds for observed and unobserved data, respectively. $\mathcal{I}^O$ and $\mathcal{I}^U$ represent the sets of observed and unobserved data indices, and $L(f_c)$ and $L(f_o)$ correspond to the sets of layers included in f_c and f_o,

[1] This case is referred to as "semi-described" [10].

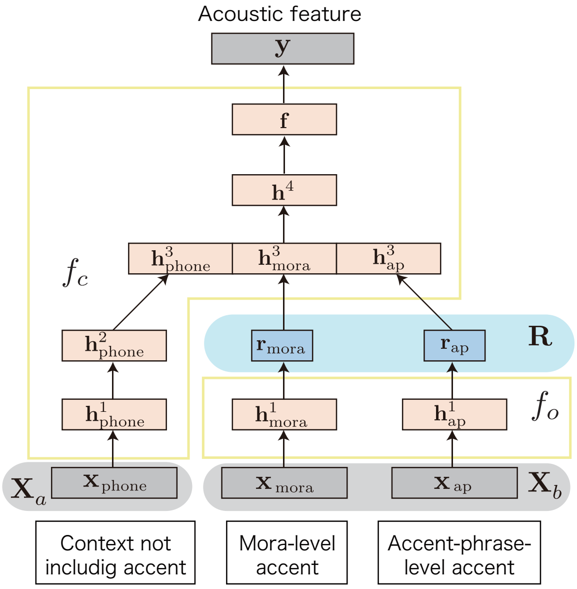

Acoustic feature
$\mathbf{y}$
$\mathbf{f}$
$\mathbf{h}^4$
$\mathbf{h}^3_{\text{phone}}$
$\mathbf{h}^3_{\text{mora}}$
$\mathbf{h}^3_{\text{ap}}$
f_c
$\mathbf{h}^2_{\text{phone}}$
$\mathbf{r}_{\text{mora}}$
$\mathbf{r}_{\text{ap}}$
$\mathbf{R}$
$\mathbf{h}^1_{\text{phone}}$
$\mathbf{h}^1_{\text{mora}}$
$\mathbf{h}^1_{\text{ap}}$
f_o
$\mathbf{X}_a$
$\mathbf{x}_{\text{phone}}$
$\mathbf{x}_{\text{mora}}$
$\mathbf{x}_{\text{ap}}$
$\mathbf{X}_b$
Context not includig accent
Mora-level accent
Accent-phrase-level accent

Figure 2: *DGP network architecture using the latent variable representation of accent. The functions and variables are categorized according to Fig. 1.*

respectively. $q(\mathbf{r}_i) = \mathcal{N}(\mathbf{r}_i; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is the variational distribution of latent variable for the i-th unobserved data point. $p(\mathbf{r}_i)$ represents the prior defined by a standard normal distribution $\mathcal{N}(\mathbf{r}_i; \mathbf{0}, \mathbf{I})$.

$\mathcal{L}^{\mathcal{O}}$ is equivalent to the ELBO of the DSVI-DGP-based regression [14]. The difference between the partial lower bounds $\mathcal{L}^{\mathcal{O}}$ and $\mathcal{L}^{\mathcal{U}}$ is that $\mathcal{L}^{\mathcal{O}}$ includes the penalty term of KL divergence accompanied by the layers of f_o, while $\mathcal{L}^{\mathcal{U}}$ includes the KL divergence for the latent variable distributions. Since both $\mathcal{L}^{\mathcal{O}}$ and $\mathcal{L}^{\mathcal{U}}$ can be represented by the sum of respective data points, we can train the parameters of DGP and latent variables by gradient-based stochastic optimization.

Although the parameters of DGP and latent variables can be trained simultaneously, the training does not always work well if we initialize the parameters of latent variable distributions with random values. Hence, we perform semi-supervised learning on the basis of the technique for single-layer GP [10] as follows:

1. Train DGP parameters using the observed data.
2. Train the parameters of latent variables for the unobserved data.
3. Train the parameters of DGP and latent variables simultaneously.

4. DGP-based Japanese speech synthesis with latent variable representation

In this study, we apply the semi-supervised learning to Japanese speech synthesis of Tokyo dialect. The modeling of pitch accent is important for Japanese speech synthesis because incorrectly uttered accent may change the meaning of a word. In Japanese speech, pitch accent is denoted in two ways: mora-level high/low expression and accent-phrase-level accent type which represents the position of the accent nucleus. To reduce the annotation cost of prosodic labels, we suppose that the accent type and high/low information is unobserved. Alternatively, we introduce two latent variable spaces of mora- and accent-phrase levels.

The network architecture of DGP model is shown in Fig. 2. The input context of observed data is partitioned into three parts: mora-level accent information $\mathbf{x}_{\text{mora}}$, accent-phrase-level accent information $\mathbf{x}_{\text{ap}}$, and the other contexts $\mathbf{x}_{\text{phone}}$ which are not related to accent and mainly consist of phonetic information. For unobserved data, we assume that $\mathbf{x}_{\text{mora}}$ and $\mathbf{x}_{\text{ap}}$ are missing and replace them by latent variables $\mathbf{r}_{\text{mora}}$ and $\mathbf{r}_{\text{ap}}$. The hidden layer variables $\mathbf{h}^3_{\text{phone}}$, $\mathbf{h}^3_{\text{mora}}$, and $\mathbf{h}^3_{\text{ap}}$ are predicted independently and are concatenated to infer the variable $\mathbf{h}^4$. We share the latent variables $\mathbf{r}_{\text{mora}}$ and $\mathbf{r}_{\text{ap}}$ in the same mora and in the same accent phrase, respectively.

5. Experiments

5.1. Experimental conditions

We used the Japanese speech data of a female speaker F009 included in the XIMERA corpus [15]. The training and test data consisted of 1533 and 60 utterances, which were approximately 119 and 4.1 minutes, respectively. The 1434 utterances, which are 90% of training data, were regarded as the partially unobserved data whose accent labels were omitted. The 99 and 60 utterances not included in the unobserved data were used as the fully observed and validation data sets, respectively. We extracted F0, spectral envelope, and aperiodicity using STRAIGHT [16] every 5 ms from the speech signal at a sampling rate of 16 kHz and obtained 0-39th mel-cepstrum, log F0, and 5-band aperiodicity. We used a 139-dimensional vector consisting of Δ, Δ^2, and voiced/unvoiced flags as acoustic features. The acoustic features are normalized to zero mean and unit variance. The dimensions of context vectors $\mathbf{x}_{\text{phone}}$, $\mathbf{x}_{\text{mora}}$, and $\mathbf{x}_{\text{ap}}$ were 477, 38, and 99, respectively, and each dimension was normalized to the range $[-1, 1]$. In this study, we assumed the condition that phrase boundaries were already given and accent types were unknown for the unobserved data.

The DGP model consisted of five layers as shown in Fig. 2, and it consisted of five layers. The dimensionalities of latent variables and other hidden layer variables were 3 and 32, respectively. The number of inducing points for each layer was set to 1024. We used ArcCos kernel [17] with the normalization process that was shown to be effective in the previous study [18]. The parameters of DGP were initialized by the pretraining method of DGP using layer-wise GP training [18]. The sampling size S was set to unity.

In the model training, we used Adam optimization [19] whose leaning rate was 0.01 and minibatch size was set to 1024. The first and third phases of training process shown in Sect. 3 were performed up to 30 and 100 epochs, respectively. In the second phase, which determines latent variables, we performed optimization for each utterance stopped when the lower bound converged or it reached 5000 epochs.

We compared the proposed semi-supervised learning with the supervised learning of DGP models using the following data sets:

FULL: 1533 utterances were used and all the prosodic labels were fully available. This is an ideal situation.

LABELED: 99 fully labeled utterances were used for training.

W/O ACCENT: 1533 utterances were used but accent information was not used as the contexts.

Table 1: *Acoustic feature distortions between original and synthetic speech. MCEP: mel-cepstral distortion [dB], F0: RMSE of log F0 [cent], DUR: RMSE of phone duration [ms].*

Method	MCEP	F0	DUR
FULL	4.79	167	16.0
LABELED	5.54	228	23.5
W/O ACCENT	4.75	207	16.1
PROPOSED	4.76	178	16.2

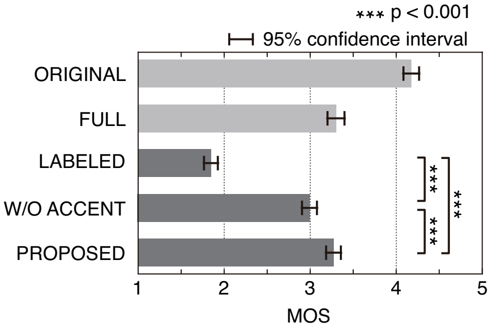


Figure 3: *Subjective evaluation results on naturalness.*

PROPOSED: 1533 utterances were used but only 99 fully labeled utterances were included.

The training of conventional methods was run up to 130 epochs, respectively.

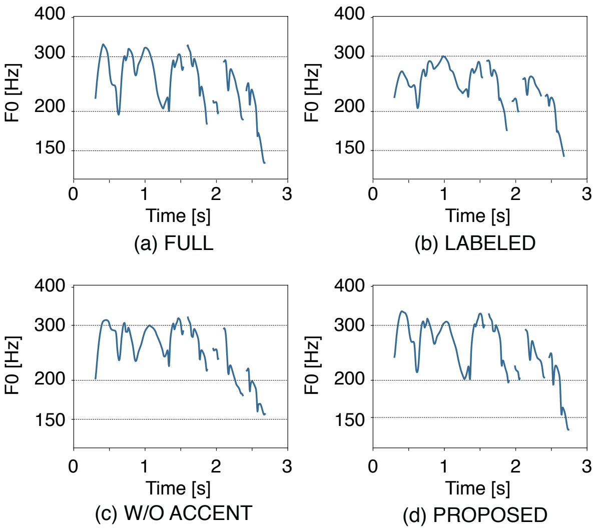
5.2. Objective evaluation

To evaluate the effectiveness of the proposed semi-supervised method, we compared the distortions between generated and original speech parameters. We used mel-cepstral distance (MCD) and the root mean squared errors (RMSEs) of log F0 and phone duration. The results are shown in Table 1. It can be seen from the results that PROPOSED yielded comparable MCD and duration distortion with FULL and W/O ACCENT. By comparing the RMSEs of log F0, whereas the distortion of W/O ACCENT was 30 cent larger than FULL, PROPOSED reduced the RMSE by 19 cent. The results indicate that the proposed semi-supervised method is effective in F0 generation. Also, we confirmed that LABELED, which uses only 99 utterances, yields larger distortions than the other methods.

5.3. Subjective valuation

We performed a listening test to evaluate the perceptual quality of synthesized speech[2]. The number of participants was 45 and the participants on crowd-sourcing service evaluated the naturalness of speech samples in a five-point scale: 5: excellent, 4: good, 3: fair, 2: poor, and 1: bad. The experiments were performed using webMUSHRA [20][3], and four sentences with respective methods were randomly chosen for each participant.

The mean opinion score (MOS) of each method is shown in Fig. 3. In the figure, ORIGINAL means the speech samples obtained directly from the original recording. It can be seen from the figure that the score of PROPOSED was comparable with FULL, which is an ideal situation. Moreover, we performed a paired t-test with significance level $\alpha = 0.05$, and found that the proposed method gave a significantly higher score than both LABELED and W/O ACCENT.



(a) FULL

(b) LABELED

(c) W/O ACCENT

(d) PROPOSED

Figure 4: *Generated F0 contours of the sentence "ii aji no wain o uru mise nara kyaku ga afureru" in Japanese, which means "Shops that sell good taste wine are filled with customers."*

5.4. Comparison of F0 contours

To see the performance differences among the methods in F0 generation, we show F0 contours generated using respective methods in Fig. 4. We can find that the proposed semi-supervised method gave a similar F0 contour with FULL. In contrast, the peaks of F0 contours of W/O ACCENT were lower than those of FULL. A possible reason is that F0 contours of W/O ACCENT are flatten since we do not use accent information, which expresses the position of F0 peak position.

6. Conclusions

In this paper, we have proposed the semi-supervised training method for speech synthesis framework based on DGP. To lessen the annotation cost of speech data, we attempted to solve the problem that the prosodic labels of speech data are not fully annotated. In the proposed method, we represent the prosodic information of the unannotated speech as latent variables which are used as the input of DGP. The subjective evaluation results, in which the prosodic labels of 90% of training data are not annotated, showed that the proposed method gave a comparable score in naturalness with the method using fully-annotated data.

For future work, we will use more diverse speech data such as tone languages and stress accent languages. Moreover, we should examine the effectiveness in low-resource languages including dialects, which are difficult to perform prosody annotation.

7. Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 17K12711.

[2] Synthetic speech samples are available at `https://hyama5.github.io/DGPLVM_prosody`.

[3] webMUSHRA includes not only MUSHRA test but also MOS and preference tests

8. References

[1] C.-Y. Yang, Z.-H. Ling, H. Lu, W. Guo, and L.-R. Dai, "Automatic phrase boundary labeling for mandarin TTS corpus using context-dependent HMM," in *7th International Symposium on Chinese Spoken Language Processing*, 2010, pp. 374–377.

[2] G.-A. Levow, "Automatic prosodic labeling with conditional random fields and rich acoustic features," in *Proc. IJCNLP*, 2008, pp. 217–224.

[3] T. Koriyama, H. Suzuki, T. Nose, T. Shinozaki, and T. Kobayashi, "Accent type and phrase boundary estimation using acoustic and language models for automatic prosodic labeling," in *Proc. INTERSPEECH*, 2014, pp. 2337–2341.

[4] C. Ding, L. Xie, J. Yan, W. Zhang, and Y. Liu, "Automatic prosody prediction for Chinese speech synthesis using BLSTM-RNN and embedding features," in *Proc.ASRU*, 2015, pp. 98–102.

[5] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[6] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2wav: End-to-end speech synthesis," in *Proc. ICLR workshop*, 2017.

[7] Y. Yasuda, X. Wang, S. Takaki, and J. Yamagishi, "Investigation of enhanced Tacotron text-to-speech synthesis systems with self-attention for pitch accent language," in *Proc. ICASSP*, 2019, pp. 6905–6909.

[8] D. Moungsri, T. Koriyama, and T. Kobayashi, "Tone modeling using Gaussian process latent variable model for statistical speech synthesis," in *Proc. Speech Prosody*, 2016, pp. 1014–1018.

[9] M. Titsias and N. D. Lawrence, "Bayesian Gaussian process latent variable model," in *Proc. AISTATS*, 2010, pp. 844–851.

[10] A. Damianou and N. Lawrence, "Semi-described and semi-supervised learning with Gaussian processes," in *Proc. UAI*, 2015, pp. 228–237.

[11] ——, "Deep Gaussian processes," in *Proc. AISTATS*, 2013, pp. 207–215.

[12] T. Koriyama and T. Kobayashi, "Statistical parametric speech synthesis using deep Gaussian processes," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 5, pp. 948–959, 2019.

[13] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT press, 2006.

[14] H. Salimbeni and M. Deisenroth, "Doubly stochastic variational inference for deep Gaussian processes," in *Proc. NIPS*, 2017, pp. 4591–4602.

[15] H. Kawai, T. Toda, J. Ni, M. Tsuzaki, and K. Tokuda, "XIMERA: A new TTS from ATR based on corpus-based technologies," in *Fifth ISCA Workshop on Speech Synthesis*, 2004.

[16] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3-4, pp. 187–207, 1999.

[17] Y. Cho and L. K. Saul, "Kernel methods for deep learning," in *Proc. NIPS*, 2009, pp. 342–350.

[18] T. Koriyama and T. Kobayashi, "A training method using DNN-guided layerwise pretraining for deep Gaussian processes," in *Proc. ICASSP*, 2019, pp. 4785–4789.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[20] M. Schoeffler, S. Bartoschek, F.-R. Stöter, M. Roess, S. Westphal, B. Edler, and J. Herre, "webMUSHRA—a comprehensive framework for Web-based listening tests," *Journal of Open Research Software*, vol. 6, no. 1, p. 8, 2018.