Interspeech 2022
18-22 September 2022, Incheon, Korea

A Unified Accent Estimation Method Based on Multi-Task Learning for Japanese Text-to-Speech

Byeongseon Park, Ryuichi Yamamoto, Kentaro Tachibana

LINE Corp., Tokyo, Japan
{park.byeongseon, ryuichi.yamamoto, kentaro.tachibana}@linecorp.com

Abstract

We propose a unified accent estimation method for Japanese text-to-speech (TTS). Unlike the conventional two-stage methods, which separately train two models for predicting accent phrase boundaries and accent nucleus positions, our method merges the two models and jointly optimizes the entire model in a multi-task learning framework. Furthermore, considering the hierarchical linguistic structure of intonation phrases (IPs), accent phrases, and accent nuclei, we generalize the proposed approach to simultaneously model the IP boundaries with accent information. Objective evaluation results reveal that the proposed method achieves an accent estimation accuracy of 80.4%, which is 6.67% higher than the conventional two-stage method. When the proposed method is incorporated into a neural TTS framework, the system achieves a 4.29 mean opinion score with respect to prosody naturalness.

Index Terms: Accent estimation, multi-task learning, accent sandhi, text-to-speech, Japanese

1. Introduction

Accent estimation plays an essential role in Japanese text-to-speech (TTS). Because Japanese is a pitch-accented language, TTS systems need to estimate accurate accent information from text [1]. Although recent end-to-end TTS has enabled high-quality TTS while eliminating the need for a complicated text processing front-end [2, 3], Japanese TTS systems still require accent estimation as a pre-processing for synthesizing natural speech [4–6].

In the case of Japanese, the accent is represented by high (H) or low (L) in the pitch of each mora as depicted in Figure 1. An utterance consists of groups of intonation phrases (IPs). The IP is the basic unit of intonation and is composed of several accent phrases (APs), where each AP contains a pitch-downstep of at most one mora. The mora where the pitch-downstep happens is called the accent nucleus (AN). The AN is considered to be as a word-level lexical attribute. However, it often changes depending on context (e.g., how words are combined). Figure 1 illustrates this phenomenon (called "accent sandhi").

To estimate the correct accent from text, conventional methods typically adopt two-step approaches that independently predict AP boundaries and AN positions using either rule-based [7, 8] or statistical model based methods [9, 10]. In particular, statistical model based approaches using conditional random fields (CRFs) [11] achieve better accent estimation accuracy by modeling the temporal dependencies of accent. However, because two CRFs are independently trained for predicting AP boundaries and AN positions, these two models cannot exploit the shared context information that is useful for predicting accurate accent.

In this paper, we propose a unified accent estimation method that jointly models the AP boundaries and AN positions using neural networks. Furthermore, considering the hierarchical linguistic structure of IPs, APs, and ANs, we generalize the proposed approach to simultaneously model the IP boundaries with accent information. Because our entire model is jointly optimized within a multi-task learning framework (i.e., solving multiple tasks with a single model) [12], it is able to exploit a general representation to predict more accurate IP, AP, and AN information.

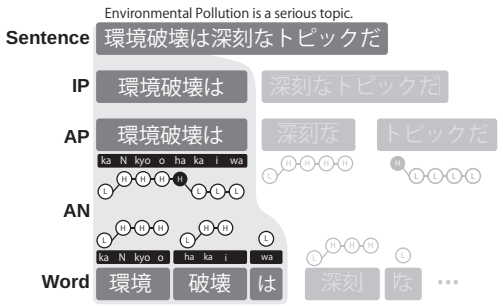

Figure 1: *Example of linguistic structure and accent sandhi in Tokyo dialect when the three words "環境" (kankyō; environmental), "破壊" (hakai; pollution) and "は" (wa; is) are combined as "環境破壊は" (kankyōhakaiwa; environmental pollution is). H, L, and the shaded circle represent the high pitch, and low pitch, and AN, respectively. Note that the Japanese accent does not necessarily contain an accent nucleus (e.g., 環境, 破壊).*

We verify that our proposed method achieves accent estimation accuracy of 80.4%, which is 6.67% higher than that of the CRF-based conventional method [10]. Furthermore, we find that the synthetic speech with our joint estimation method has improved naturalness with respect to prosody[1].

2. Related work

Multi-task learning based approaches are not new, and the effectiveness of the approaches has been verified on different NLP tasks, such as named entity recognition [13], question answering [14], and the front-end of TTS system [15–17]. However, as discussed by Standley et al. [18], the overall performance of multi-task learning depends on the relationship between the jointly trained tasks. Therefore, to maximize the performance of accent estimation within the multi-task learning framework, we propose to jointly solve three tasks by considering the hierarchical linguistic structure of IP, AP, and AN.

In addition, there are other studies that estimate accent from text. For example, a neural network based model is proposed to expand the dictionary for the Japanese accent [19]. In this method, the model estimates the accent as a sequence of three states, upstep, downstep, and neutral, from surface and phonetic

[1] In this paper, we evaluate the naturalness based on the Tokyo dialect (i.e., the standard Japanese)

10.21437/Interspeech.2022-334

information. Because this method primarily focuses on word accent estimation, it cannot estimate the accent sandhi for words not in a dictionary. Hida et al. [20] recently proposed to use a pre-trained language model [21] for polyphone disambiguation and accent estimation. However, their method does not consider the relationship between AP and AN tasks during the training process.

Other works have attempted to simultaneously estimate the accent with other linguistic information such as phonemes and word boundaries [22, 23]. In particular, a neural machine translation model based on the Transformer [24] enables the accurate estimation of multiple target information from text [23]. However, such models tend to require a large amount of training data (e.g., 5 million sentences) to model complex relationship between text and multiple targets including phonemes.

In contrast with those existing methods, our method properly handles accent sandhi by modeling the temporal dependencies of IP, AP and AN simultaneously. Furthermore, because we focus on the accent estimation task, our model can be trained with a relatively small volume of data (e.g., less than 0.1 million sentences).

3. Method

3.1. Task definition

In this study, we formulate the accent estimation task as a sequence labeling problem. Because the APs have linguistic dependencies with IPs, we include the IP boundary prediction in the accent estimation task in the methods presented in the rest of our paper.

We assume the input linguistic features $\mathbf{x} = [x_1, x_2, ..., x_N]^T$, where N denotes the number of moras, are given. Our goal is to estimate the target label sequences for the IPs, APs, and ANs: $\mathbf{y}^{\mathrm{IP}} = \left[y_1^{\mathrm{IP}}, y_2^{\mathrm{IP}}, ..., y_N^{\mathrm{IP}}\right]^T$, $\mathbf{y}^{\mathrm{AP}} = \left[y_1^{\mathrm{AP}}, y_2^{\mathrm{AP}}, ..., y_N^{\mathrm{AP}}\right]^T$, and $\mathbf{y}^{\mathrm{AN}} = \left[y_1^{\mathrm{AN}}, y_2^{\mathrm{AN}}, ..., y_K^{\mathrm{AN}}\right]^T$, respectively, where K is the number of APs. Each target label is defined as follows:

$$y_n^{\mathrm{IP}} \in \{0, 1\}$$
$$y_n^{\mathrm{AP}} \in \{0, 1\}$$
$$y_k^{\mathrm{AN}} \in \{0, 1, ..., M\}$$

where the binary value in the IP/AP labels represent whether the n-th mora is a phrase boundary or not, the numbers in the AN label represent the AN's location in the k-th AP, and M represents the maximum number of the AN's locations. We use $y_k^{\mathrm{AN}} = 0$ as a special AN label representing that the AP does not contain an AN. The number M is set at 20 according to the statistics of our dataset. Note that the input features, IP labels, and AP labels are defined at mora-level, whereas the AN labels are defined at AP-level [2].

3.2. Pre-processing

To obtain mora-level linguistic features $\mathbf{x}$, we tokenized the input text and extracted the word-level features using the Japanese text analyzer MeCab [25] and the Japanese dictionary Unidic [26]. The details of the word-level features are as follows: (1) surface (word surface); (2) part-of-speech (POS) tag; (3) goshu (word category, e.g., a Japanese word or loan-word); (4) cType (word conjugation type); (5) cForm (word conjugation form); (6) aType (location of the AN when read as a single word); (7) aConType (accent sandhi type); (8) aModType (accent sandhi type depending on the cForm). The word-label features were expanded to mora-level features and used as the input of our model.

[2] One can represent ANs as mora-level labels [19] or phoneme-level labels [23]. However, these approaches cannot guarantee the linguistic restriction that an AP contains at most one AN [10].

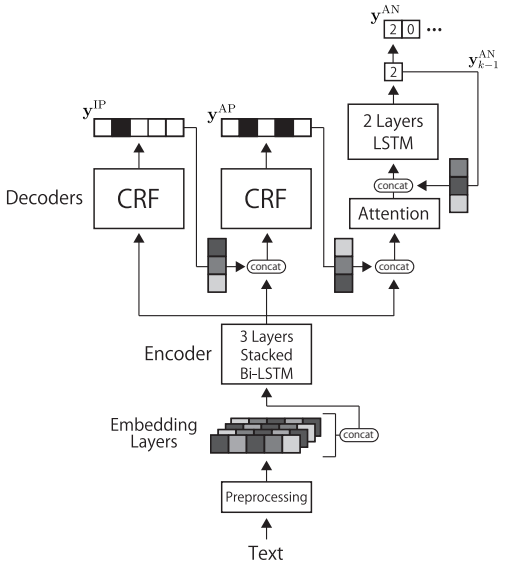

Figure 2: *Overview of the proposed model. The model consists of embedding layers, an encoder, and three decoders for predicting IP, AP, or AN labels.*

3.3. Model

Let $Y = \{\mathbf{y}^{\mathrm{IP}}, \mathbf{y}^{\mathrm{AP}}, \mathbf{y}^{\mathrm{AN}}\}$ be the set of all target labels. We model the conditional joint distribution of target labels based on the hierarchical linguistic relationship of IPs, APs, and ANs (as depicted in Figure 1), as follows:

$$p(Y|\mathbf{x}) = p(\mathbf{y}^{\mathrm{AN}}|\mathbf{y}^{\mathrm{AP}}, \mathbf{x})p(\mathbf{y}^{\mathrm{AP}}|\mathbf{y}^{\mathrm{IP}}, \mathbf{x})p(\mathbf{y}^{\mathrm{IP}}|\mathbf{x}), \quad (1)$$

where each term represents a conditional probability distribution for the AN, AP, and IP labels, respectively. In our method, the conditional distributions are parameterized using neural networks.

Figure 2 shows an overview of the proposed model. The input categorical linguistic features that were extracted from text are first converted to a continuous representation using a set of embedding layers. All the embeddings are then concatenated and converted to hidden context features by the encoder, which consists of three stacked bi-directional long short-term memory (Bi-LSTM) layers that model the temporal dependencies of text. The output of the encoder is used as the input for the decoders.

The decoders predict the conditional probability of the target label sequence given the encoded context features. To model the conditional dependencies in Equation (1), the output of the decoders for IPs and APs are concatenated with the encoder output to compose the input of the decoders for APs and ANs, respectively. For the IPs and APs, the decoders use CRFs to compute the output probability. In addition, the decoder for the ANs employs an autoregressive (AR) model. To align mora-level features to AP-level predictions while considering temporal correlations of ANs, the decoder for the ANs uses two uni-directional LSTM layers with attention mechanism [27].

In the training stage, the entire model is optimized to maximize the log-likelihood given the training data. The log-likelihood can be computed as a sum of three log probabilities based on Equation (1). It should be noted that maximizing the

Table 1: *Objective evaluation results on each task. F_1 and Accuracy are mora-level F_1 score and sentence-level accuracy, respectively.*

				IP		AP		AN	
Model	**Encoder**	**Decoder**	**Task**	**F_1**	**Accuracy**	**F_1**	**Accuracy**	**F_1**	**Accuracy**
(a) [10]	-	CRF	AP	-	-	98.67%	90.16%	-	-
(b) [10]	-	CRF	AN	-	-	-	-	97.00%	74.51%
(c)	Bi-LSTM	CRF	IP	95.18%	**78.44%**	-	-	-	-
(d)	Bi-LSTM	CRF	AP	-	-	99.28%	90.77%	-	-
(e)	Bi-LSTM	AR	AN	-	-	-	-	95.04%	78.29%
(f)	Bi-LSTM	CRF, AR	AP+AN	-	-	99.25%	90.62%	96.95%	79.49%
(g)	Bi-LSTM	CRF, CRF, AR	IP+AP+AN	**95.20%**	78.31%	**99.30%**	**90.96%**	**97.33%**	**80.98%**

Table 2: *Comparison on each task combination. The scores are the sentence-level accuracy on multiple tasks.*

System	**Components**	**IP+AP+AN**	**IP+AP**	**AP+AN**
(A)	(a), (b), (c)	59.89%	72.16%	73.73%
(B)	(c), (d), (e)	61.82%	72.42%	75.36%
(C)	(c), (f)	64.64%	72.73%	79.00%
(D)	(g)	**65.58%**	**72.82%**	**80.40%**

log-likelihood is equivalent to simultaneously solving the three tasks (i.e., IP, AP and AN predictions) while sharing the embedding and the encoder parameters. In the prediction stage, the optimal target labels can be computed by:

$$\hat{Y} = \arg\max_{Y} p(Y|\mathbf{x}). \tag{2}$$

However, because it is computationally expensive to search all possible combinations of target labels [28], we predict the IP, AP and AN labels in order, as follows:

$$\hat{\mathbf{y}}^{\mathrm{IP}} = \arg\max_{\mathbf{y}^{\mathrm{IP}}} p(\mathbf{y}^{\mathrm{IP}}|\mathbf{x}), \tag{3}$$

$$\hat{\mathbf{y}}^{\mathrm{AP}} = \arg\max_{\mathbf{y}^{\mathrm{AP}}} p(\mathbf{y}^{\mathrm{AP}}|\hat{\mathbf{y}}^{\mathrm{IP}}, \mathbf{x}), \tag{4}$$

$$\hat{\mathbf{y}}^{\mathrm{AN}} = \arg\max_{\mathbf{y}^{\mathrm{AN}}} p(\mathbf{y}^{\mathrm{AN}}|\hat{\mathbf{y}}^{\mathrm{AP}}, \mathbf{x}). \tag{5}$$

Though this approach is sub-optimal, we empirically find it works well.

4. Experiments

4.1. Experimental setup

4.1.1. Database

For the experiments, we collected the recorded speech of approximately 89,061 Japanese utterances. The prosody information (i.e., IPs, APs, ANs) was manually annotated by a linguistic expert. We split the dataset into three subsets of 80,061, 4,500, and 4,500 utterances for the training, validation, and test sets, respectively.

4.1.2. Model details

To evaluate the proposed method, we investigated seven models with different task configurations, as listed in Table 1. As the baseline system, we used two CRF-based models for predicting APs and ANs (i.e., model (a) and (b)) [10]. For the other five models, we prepared three groups as follows: (1) the single-task based models, i.e., model (c) to (e); (2) the dual-task based models, i.e., model (f); (3) the triple-task based model, i.e., model (g). The model (c) to (g) used the same architecture except for the decoders, where the number of decoders was adjusted by the target tasks. For the model (c) to (g), we set the embedding sizes for the input linguistic features as follows: (1) mora: 256, (2) surface: 512, (3) POS tag: 128, (4) goshu: 64, (5) cType: 256, (6) cForm: 128, (7) aType: 64, (8) aConType: 64, (9) aModType: 64. Note that we heuristically determined the embedding sizes. The encoder consists of three Bi-LSTM layers with 512 dimensions. For the decoders of the APs and ANs, we used the embedding layer for the hierarchical conditioning, setting the decoder output embedding size for IPs and APs to 32, with a dropout $p = 0.5$. The decoder for ANs has two uni-directional LSTM layers with 512 dimensions. This decoder also has an embedding layer to incorporate the previous output of the decoder, with an embedding size of 32 and a dropout $p = 0.5$. In the training stage, the decoders were trained with the teacher forcing. We trained models (c) to (g) for 30 epochs with a mini-batch size of 128 utterances. We used the Adam optimizer [29]: the parameters were $(\alpha, \beta_1, \beta_2, \varepsilon) = (10^{-3}, 0.9, 0.99, 10^{-7})$. The baseline models were trained using the publicly available code[3] by the authors of [10].

4.2. Objective evaluation

Tables 1 and 2 shows the sentence-level accent estimation accuracy on a single-task basis and multiple-task basis, respectively. The findings are summarized as follows.

Single task vs. Multiple tasks A comparison of models (a)–(e) with models (f) and (g) in Table 1 and systems (A) and (B) with systems (C) and (D) in Table 2, provides evidence of the improvement in performance achieved by the multi-task learning framework. As shown in Table 1, models (d) and (e) achieved sentence-level accuracies of 90.77% and 78.29% on the **AP** and **AN** tasks, respectively. However, when the models are combined as system (B), the accuracy on AP+AN is reduced to 75.36%. The gap (-2.93%) can be regarded as the decrease caused by the discrepancy between the single-task based models. On the other hand, model (g) (i.e., system (D) in Table 2) achieved the sentence accuracy of 80.98% and 80.40% on the **AN** and **AP+AN** tasks, respectively, and the decrease in performance has been substantially reduced to -0.58%. This confirms that the multi-task learning framework works to learn a more general representation by optimizing multiple tasks simultaneously.

Baseline vs. Proposed model As shown in Table 1, our proposed model (i.e., model (g)) outperformed the baseline models [10] (i.e., models (a) and (b)) on all tasks. Moreover, our

[3] https://sites.google.com/site/suzukimasayuki/accent

Table 3: *MOS test results with 95% confidence intervals. Note that Reference denotes the speech samples synthesized using human-annotated prosody information.*

System	MOS
(A)	4.15 ± 0.06
(B)	4.15 ± 0.06
(C)	$\mathbf{4.32 \pm 0.05}$
(D)	4.29 ± 0.06
Reference	4.47 ± 0.04

single-task based models (i.e., models (d) and (e)) achieved higher scores than the baseline models. This confirms that the combination of the multi-task learning framework and the Bi-LSTM encoder is more effective to improve the performance of accent estimation.

Effectiveness of task extension In addition, we compared model (f) and (g) to analyze the effectiveness on the task extension of IPs. Table 1 shows that model (g) outperformed model (f) on all tasks. This result confirms that the task extension is beneficial to the generalization of the model, as conformed in the other multi-task learning based approaches [30, 31].

4.3. Application to TTS

4.3.1. TTS setup

To verify the effectiveness of the proposed method in TTS scenarios, we integrated the proposed method into the text processing front-end of a FastSpeech 2 based TTS system [32]. Audio samples are available online[4].

The TTS system consists of two models: (1) a feed-forward Transformer-based acoustic model that predicts acoustic features from a phoneme sequence with additional prosody information (i.e., IPs, APs, and ANs), and (2) a Parallel WaveGAN vocoder that generates speech waveforms from acoustic features [33]. The detailed model structure and training conditions of these two models were the same as those in [34]. As a database, we used the same corpus described in the Section 4.1.1. The speech corpus was recorded by a single Japanese professional speaker. The speech signals were sampled at 24 kHz, and each sample was quantized to 16 bits. The total amount of the training data size was 116 hours.

4.3.2. Subjective listening tests

To analyze the performance of TTS systems with the proposed accent estimation, naturalness mean opinion score (MOS) listening tests were performed. For the test, we compared the four systems in Table 2. We randomly selected 70 sentences from the test set. We then synthesized the speech samples using the TTS system with the predicted prosody information for each system. Furthermore, we also synthesized the speech using the human-annotated prosody information. In total, we used 350 utterances in the test. We asked 17 native Japanese speakers to judge the naturalness of the prosody of speech samples with the following five-scale responses: 1 = Bad; 2 = Poor; 3 = Fair; 4 = Good; and 5 = Excellent. Note that we showed the texts of the speech samples to the raters to help accurately judge the naturalness of prosody.

Table 3 shows the performance on different systems. The results confirm that the the multi-task learning based methods (i.e., systems (C) and (D)) achieved significantly higher naturalness than the single-task based methods, including CRF-based conventional method (i.e., system (A)) and neural network based method (i.e., system (B)).

[4] https://6gsn.github.io/demos/mtl_accent/

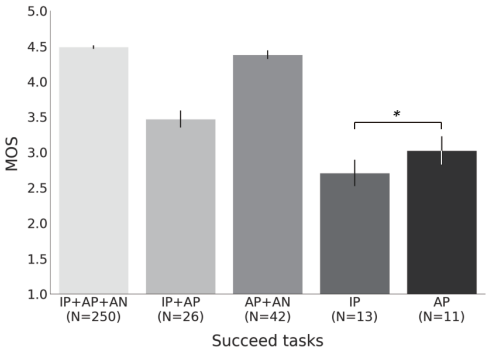

Figure 3: *MOS test results for each succeeded task combination.* N*: the number of samples, *: small significant differences with* $p \leq .05$*, no tag: highly significant differences with* $p \leq .001$*. Note that* p *is the p-value of the Mann–Whitney* U*-test statistic.*

Furthermore, Figure 3 presents the naturalness for each succeeded task configuration. The labels in Figure 3 denote which tasks succeed for each sample in the test set. For example, **IP** refers to the sample synthesized using correct IPs with the wrong APs and ANs. Note that we omit the three groups of the samples from Figure 3 as follows: (1) **IP+AN**, (2) **AN**, (3) **X** (i.e., failed on all tasks) because the number of samples was not enough (i.e., two, three, three, respectively) to find statistical significance compared to other types of samples. The results showed the following significant facts: (1) the samples for which the models were able to complete the all tasks (i.e., **IP+AP+AN**) have a higher naturalness than others; (2) the samples for which the models were able to complete the **AN** tasks (i.e., **IP+AP+AN** and **AP+AN**) have a higher naturalness; (3) the samples for which the models were able to complete the **IP+AP** tasks have higher naturalness than those for which only the single task (i.e., **IP** and **AP**) could be completed. The result confirms that the AN task had the most significant impact on perceptual quality. Furthermore, the result also demonstrated that a model that can succeed not only on a single task but also on as many multiple tasks as possible is more suitable as a front-end for TTS.

5. Conclusion

We proposed a novel Japanese accent estimation method based on multi-task learning. We merged the models that are required to estimate the Japaneses accent and jointly optimized the overall model. In the objective evaluation, the proposed method achieved a sentence-level accuracy of 80.4% on accent estimation; this accuracy is 6.67% higher than that of the baseline system [10]. Moreover, subjective evaluations in the TTS scenarios verified that the proposed method achieved significantly better perceptual quality than the conventional single-task based methods. Future work includes utilizing a learned representation on massive unlabeled data to improve the accent estimation accuracy.

6. Acknowledgments

This work was supported by Clova Voice, NAVER Corp., Seongnam, Korea. The authors would like to thank Yuma Shirahata and Kosuke Futamata at LINE Corp., Tokyo, Japan, for their support.

7. References

[1] H.-T. Luong, X. Wang, J. Yamagishi, and N. Nishizawa, "Investigating accuracy of pitch-accent annotations in neural network-based speech synthesis and denoising effects," in *Proc. Interspeech*, 2018, pp. 37–41.

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

[3] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, 2018, pp. 4779–4783.

[4] T. Fujimoto, K. Hashimoto, K. Oura, Y. Nankaku, and K. Tokuda, "Impacts of input linguistic feature representation on Japanese end-to-end speech synthesis," in *Proc. SSW*, 2019, pp. 166–171.

[5] 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. 302–311.

[6] K. Kurihara, N. Seiyama, and T. Kumano, "Prosodic features control by symbols as input of sequence-to-sequence acoustic modeling for neural TTS," *IEICE Transactions on Information and Systems*, vol. E104-D, no. 2, pp. 302–311, 2021.

[7] Y. Sagisaka and H. Sato, "Accentuation rules for Japanese word concatenation," *IEICE Transactions on Information and Systems*, vol. J66-D, no. 7, pp. 849–856, 1983.

[8] R. Kita, N. Minematsu, and K. Hirose, "Development of rules of word accent sandhi and their improvement for Japanese TTS systems," IEICE, Tech. Rep., 2002.

[9] N. Minematsu, S. Kobayashi, S. Shimizu, and K. Hirose, "Improved prediction of Japanese word accent sandhi using CRF," in *Proc. Interspeech*, 2012, pp. 2562–2565.

[10] M. Suzuki, R. Kuroiwa, K. Innami, S. Kobayashi, S. Shimizu, N. Minematsu, and K. Hirose, "Accent sandhi estimation of Tokyo dialect of Japanese using conditional random fields," *IEICE Transactions on Information and Systems*, vol. E100-D, no. 4, pp. 655–661, 2017.

[11] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. ICML*, 2001, pp. 282–289.

[12] R. Caruana, "Multitask learning: A knowledge-based source of inductive bias," in *Proc. ICML*, 1993, pp. 41–48.

[13] T.-H. Pham, K. Mai, N.-M. Trung, N.-T. Duc, D. Bolegala, R. Sasano, and S. Sekine, "Multi-task learning with contextualized word representations for extented named entity recognition," *arXiv preprint arXiv:1902.10118*, 2019.

[14] S. Joty, L. Marquez, and P. Nakov, "Joint multitask learning for community question answering using task-specific embeddings," in *Proc. EMNLP*, 2018, pp. 4196–4207.

[15] Y. Huang, Z. Wu, R. Li, H. Meng, and L. Cai, "Multi-task learning for prosodic structure generation using BLSTM RNN with structured output layer," in *Proc. Interspeech*, 2017, pp. 779–783.

[16] Y. Zheng, J. Tao, Z. Wen, and Y. Li, "BLSTM-CRF based end-to-end prosodic boundary prediction with context sensitive embeddings in a text-to-speech front-end," in *Proc. Interspeech*, 2018, pp. 47–51.

[17] H. Pan, X. Li, and Z. Huang, "A mandarin prosodic boundary prediction model based on multi-task learning," in *Proc. Interspeech*, 2019, pp. 4485–4488.

[18] T. Standley, A. Zamir, D. Chen, L. Guibas, J. Malik, and S. Savarese, "Which tasks should be learned together in multi-task learning?" in *Proc. ICML*, 2020.

[19] H. Tachibana and Y. Katayama, "Accent estimation of Japanese words from their surfaces and romanizations for building large vocabulary accent dictionaries," in *Proc. ICASSP*, 2020, pp. 8059–8063.

[20] R. Hida, M. Hamada, C. Kamada, E. Tsunoo, T. Sekiya, and T. Kumakura, "Polyphone disambiguation and accent prediction using pre-trained language models in Japanese TTS front-end," in *Proc. ICASSP (in press)*, 2022.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[22] T. Nagano, S. Mori, and M. Nishimura, "A stochastic approach to phoneme and accent estimation," in *Proc. Interspeech*, 2005, pp. 3293–3296.

[23] N. Kakegawa, S. Hara, M. Abe, and Y. Ijima, "Phonetic and prosodic information estimation from texts for genuine Japanese end-to-end text-to-speech," in *Proc. Interspeech*, 2021, pp. 126–130.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.

[25] T. Kudo, K. Yamamoto, and Y. Matsumoto, "Applying conditional random fields to Japanese morphological analysis," in *Proc. EMNLP*, 2004, pp. 230–237.

[26] Y. Den, J. Nakamura, T. Ogiso, and H. Ogura, "A proper approach to Japanese morphological analysis: Dictionary, model, and evaluation," in *Proc. LREC*, 2008, pp. 1019–1024.

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

[28] 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. 2337–2340.

[29] J. B. Diederik P. Kingma, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.

[30] J. Baxter, "A model of inductive bias learning," *Journal of Artificial Intelligence Research*, vol. 12, pp. 149–198, 2000.

[31] V. Sanh, T. Wolf, and S. Ruder, "A hierarchical multi-task approach for learning embeddings from semantic tasks," in *Proc. AAAI*, 2019, pp. 6949–6956.

[32] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text-to-speech," in *Proc. ICLR*, 2021.

[33] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020, pp. 6199–6203.

[34] R. Yamamoto, E. Song, M.-J. Hwang, and J.-M. Kim, "Parallel waveform synthesis based on generative adversarial networks with voicing-aware conditional discriminators," in *Proc. ICASSP*, 2021, pp. 6039–6043.