Interspeech 2022
18-22 September 2022, Incheon, Korea

Predicting VQVAE-based Character Acting Style from Quotation-Annotated Text for Audiobook Speech Synthesis

Wataru Nakata[1], *Tomoki Koriyama*[1], *Shinnosuke Takamichi*[1], *Yuki Saito*[1]
Yusuke Ijima[2], *Ryo Masumura*[2], *Hiroshi Saruwatari*[1]

[1]The University of Tokyo
[2]Nippon Telegraph and Telephone Corporation
nakata-wataru855@g.ecc.u-tokyo.ac.jp, t.koriyama@ieee.org

Abstract

We propose a speech-synthesis model for predicting appropriate voice styles on the basis of the character-annotated text for audiobook speech synthesis. An audiobook is more engaging when the narrator makes distinctive voices depending on the story characters. Our goal is to produce such distinctive voices in the speech-synthesis framework. However, such distinction has not been extensively investigated in audiobook speech synthesis. To enable the speech-synthesis model to achieve distinctive voices depending on characters with minimum extra annotation, we propose a speech synthesis model to predict character appropriate voices from quotation-annotated text. Our proposed model involves character-acting-style extraction based on a vector quantized variational autoencoder, and style prediction from quotation-annotated texts which enables us to automate audiobook creation with character-distinctive voices from quotation-annotated texts. To the best of our knowledge, this is the first attempt to model intra-speaker voice style depending on character acting for audiobook speech synthesis. We conducted subjective evaluations of our model, and the results indicate that the proposed model generated more distinctive character voices compared to models that do not use the explicit character-acting-style while maintaining the naturalness of synthetic speech.

Index Terms: audiobook speech synthesis, VQVAE, fictional character embedding

1. Introduction

As synthetic speech by neural text-to-speech synthesis approaches similar naturalness to human speech [1, 2, 3, 4], its applications are becoming more diverse. One such application is audiobook speech synthesis [5], which aims to synthesize diverse speech from literary books and automate audio-content production without new recordings. One challenge in audiobook speech synthesis is to learn expressive skills of professional audiobook narrators. For example, audiobook speech by professional audiobook narrators has long and complicated prosody structures unlike that by amateur speakers [6, 7]. Such prosody often reflects long-term context which includes the emotion of the character and story structure. Studies have attempted to model long-term context for improving the quality of speech produced by audiobook speech synthesis [8, 9, 10].

The narrator sometimes acts as a fictional character, which we call *character acting*. This results in the listener's better understanding of the content and makes the audiobook titles more engaging. Such character acting reflects character attributes (e.g. sex, age, etc.) and their relationship with other characters (e.g. good guys, bad guys, family etc.). Therefore, such character acting should also be present in synthesized audiobook speech and differs from emotional and expressive speech synthesis [11, 12]. However, character acting has not been extensively investigated in audiobook speech synthesis. One possible approach is to use emotional and expressive speech synthesis using explicit character-acting-style tags instead of emotion tags [13]. Also, Kato et. al. tackled similar issue in Rakugo[1] speech synthesis by annotating utterances with extensive context tags (character role, character individuality, situation, story structure etc.) [14]. However, determining comprehensive criteria for character-acting-style tagging is difficult because the acting of professional audiobook narrators is diverse. Therefore, this approach is not appropriate to achieve character acting in audiobook speech synthesis. Another possible approach is speaker adaptation from multi-speaker speech synthesis via the recording of character-acted voice [15, 16]. Also, Greene et al. [17] tackled this problem by annotating the fictional-character attributes using crowdsourcing and proposed a method of predicting the appropriate voice style for the character from those attributes. In contrast, we aim to automate audiobook creation from quotation-annotated text shown in Table 1 without such external datasets as shown in Figure 1.

We propose an audiobook speech synthesis model using character-acting styles and character embeddings. We extract character-acting styles from speech during training based on a vector quantized variational autoencoder (VQVAE) [18]. During inference, the model predicts appropriate character-acting styles for simulating a situation in which a professional audiobook narrator appropriately controls different styles for

Figure 1: *Overview of our goal to simulate human professional audiobook narrator's character acting by using our audiobook-speech-synthesis model*

Table 1: *Example of quotation-annotated texts: conversation between two insect characters*

Character	Phrase
(Narration)	The foremost ant said.
Ant	"The other day, there were chocolates, and ice cream..."
Ant girl	"Yes. Human children from school had dropped them..."
Ant	"I wonder if there were any treats today, too."

[1]Japanese traditional storytelling show

10.21437/Interspeech.2022-638

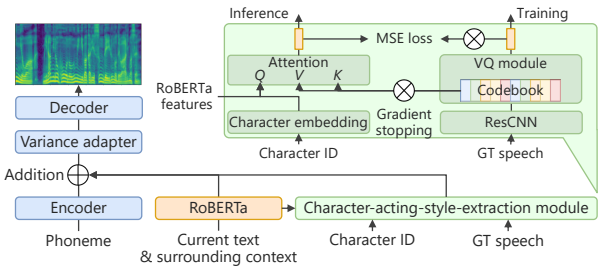

Figure 2: *Architecture of proposed model*

different fictional characters. Character embeddings are the attribute representations obtained from the character's name, conversational sentences, and surrounding characters using a similar manner to that with the Skip-Gram network [19]. Character embeddings are used to predict the character-acting styles of unseen fictional characters. The subjective evaluation results we conducted indicate that VQVAE-based acting style extraction improves the naturalness of audiobook speech. The proposed model also outperformed simple speech-synthesis models that only use texts and character names as input. The key contributions of this work are as follows:

- We propose audiobook-speech-synthesis model using character-acting styles and character embeddings and presented its effectiveness in terms of character distinction in audiobook-speech-synthesis
- We made available opensource fictional-character annotations for an audiobook corpus [8][2].

Speech samples are available online[3].

2. Proposed Model

Figure 2 shows the architecture of the proposed model. During inference, the model takes phoneme sequence, current text, surrounding context, and character ID as input. During training ground-truth speech (GT speech) is also used as input to learn character-acting style from the speech. The proposed model is based on FastSpeech2 [3]. FastSpeech2 is conditioned by character-acting styles that represents the character distinction of audiobook narrators. The character-acting style is predicted from character ID and RoBERTa [20] outputs, which include character-specific information such as frequently used words.

The RoBERTa outputs are also directly input into FastSpeech2 to model context-aware information. We model context-aware information by using not only the text corresponding to the current sentence but also its neighboring sentences. To be precise, sentence-wise embeddings are calculated by averaging subword-level ones and added to the encoder output of the FastSpeech2. It has been reported that inputting neighboring sentences makes the synthetic speech context-aware, i.e., the speech characteristics vary depending on not only the current sentence but also on the preceding and succeeding sentences. This results in the generation of appropriate synthetic speech for audiobooks [8].

[2]https://sites.google.com/site/shinnosuketakamichi/research-topics/j-kac_corpus

[3]https://wataru-nakata.github.io/is2022-audiobook/

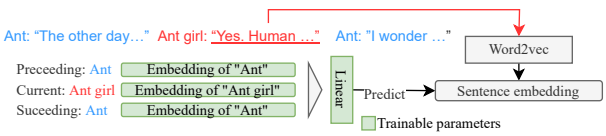

Figure 3: *Process of training of character embedding from conversational sentences proposed in [19].*

2.1. Character-acting-style-extraction module

The character-acting-style-extraction module has two functions. The first function is extracting character-acting styles from ground-truth speech using ResCNN [21] and quantizes the extracted feature vector using VQVAE [18]. One common model for speaking style extraction is a reference encoder [22]. However, our model requires modelling speaking styles of different characters acted by a single professional speaker, which is similar to a speaker-verification model that learns discriminative speaker representations. Therefore, we take ResCNN, which can capture time-invariant characteristics. It is also reported to be effective for extraction of speaker embeeding in speaker verification [21] The extracted feature vector is then fed into the VQ module to obtain one of a limited set of codes in the codebook. The quantized feature vector is finally used to condition the speech-synthesis model during training. This module efficiently learns the diverse character-acting styles expressed by a professional narrator to improve the generalization ability of the model as well as the controllability of the acting. We consider the extracted feature from this module as a character-acting style. By incorporating the VQVAE into the extraction of character-acting styles, as the training progresses, the mutual information between the output of the character-acting-style-extraction module and textual features (i.e, the phoneme feature and the RoBERTa feature) becomes maximized [18]. Hence, the output of the character-acting-style-extraction module should only contain information regarding such styles. This is why we regard its output as a character-acting style.

The second function is to predict the appropriate character-acting style from character ID and textual features. This is achieved using Character Embedding (Skip-Gram) [19], which we describe in the Section 2.2 and text features encoded using RoBERTa. Both features are concatenated and used as the query of the attention layer. For key and value input of the attention layer, we use the codebook of the VQVAE. Therefore, the output of the attention layer is a weighted sum of codes in the VQVAE codebook. During training, the difference between the output of the attention layer and the output from the VQ module is used as the loss to be minimized. During inference, the output of the attention layer is used to condition the speech-synthesis model. By doing so, the model can predict the appropriate character-acting style from RoBERTa features and the fictional character's embedding. When conducting backpropagation, we use gradient stop on the vector-quantized feature and VQVAE codebook. This is to prevent the extracted character-acting style from being easily predictable.

2.2. Extraction of fictional character's embedding

We extract character embedding from books using Character Embedding (Skip-Gram) proposed by Azab et al. [19]. This model extracts an embedding on the basis of characters' names and content of dialogues in a movie script. The extracted embedding encapsulates character-related information

such as character relatedness and improves the performance on question-answering tasks, which require understanding of dialogues. Therefore, we use this model to predict character-acting styles for audiobook speech synthesis. The architecture of Character Embedding (Skip Gram) is based on Skip-Gram of word2vec. The character embeddings to be trained are combined with the preceding and succeeding fictional character's embeddings, and word2vec-based sentence embeddings of the current character are predicted using a trainable matrix as shown in Figure 3. To apply this model to audiobook speech synthesis, we remove narrative sentences and treat the remaining sentences in the same manner as movie scripts. Hence, the character-embedding vector corresponding to the narrative style is not extracted. For narrative style, we use a zero vector as an embedding. Through this process, the learned embedding should represent information regarding the role of the character and the character's relationship with other characters.

2.3. Training criterion

We define the training criterion as follows:

$$L = L_{\text{TTS}} + \lambda_1 L_{\text{VQ}} + \lambda_2 L_{\text{attention}}, \tag{1}$$

$$L_{\text{VQ}} = \| \text{sg}[z_e(x)] - e\|_2^2 + \beta \| z_e(x) - \text{sg}[e]\|_2^2, \tag{2}$$

$$L_{\text{attention}} = \| \text{sg}[e] - \hat{e}\|_2^2 \tag{3}$$

where L_{TTS} is the loss regarding training of speech-synthesis model (i.e., error of the speech feature prediction), L_{VQ} is the loss regrading training of VQVAE, and $L_{\text{attention}}$ is the loss regarding the character-acting-style prediction. λ_1 and λ_2 are used to adjust the gain of each loss. sg is the stop gradient operation, $z_e(x)$ is the ResCNN output and e is the code that has the closest L2 distance to $z_e(x)$, and $\hat{e}$ is the predicted character-acting style.

3. Experiments

We compared the proposed model with the following models.

- FS2 (w/o BERT): Ordinary FastSpeech2.
- FS2: FastSpeech2 conditioned by cross-sentence context from RoBERTa.
- FS2-ResCNN: FS2 conditioned by ResCNN features from ground truth speech.
- FS2-ResCNN-VQ: FS2 conditioned by vector quantized ResCNN features from ground truth speech.
- FS2-character: FS2 conditioned by fictional character embeddings.
- FS2-all: Proposed model described in Section 2

Note that FS2-ResCNN and FS2-ResCNN-VQ require ground-truth speech input during inference. These models are written in red in the results presented in the following sections. For the implementation of FastSpeech2, we used the version by the first author on Github[4].

3.1. Experimental conditions

We used Japanese Kamishibai and Audiobook Corpus (J-KAC) [8] for experimental evaluations. J-KAC is composed of approximately 9 hours of audiobook speech uttered by a single male professional speaker. We downsampled the speech signals to 22.05 kHz in advance and segmented them into sentence level. We then carried out forced alignment using Julius [23] and split the J-KAC corpus into 5129, 100, and 81 utterances as training, validation, and test sets, respectively. The test set corresponded to one kamishibai (picture stories) book and had no overlap in fictional characters with other books. The frame length and frame shift for extracting an 80-dimensional mel-spectrogram were 1024 and 256 samples, respectively. As the input to the FastSpeech2 encoder, we used the sum of phoneme and accent embeddings in a similar manner to phonetic and prosodic labels [24]. For the variance adapter configurations of FastSpeech2, we used pitch, energy, and duration predictors. The phoneme-wise pitch and energy features were predicted using the variance adaptors referring to FastPitch [25] for stable training and better speech quality.

For the pretrained model of RoBERTa, we used japanese-roberta-base[5] by Rinna Co., Ltd. This model was trained on Japanese Wikipedia and the Japanese subset of CC100 [26]. As input of RoBERTa, we used current, preceding, and succeeding sentences totaling up to 510 tokens. During training of the proposed model, we froze the word embedding layer of RoBERTa because some tokens did not appear on J-KAC.

For the extraction of fictional character embeddings, we first trained word2vec using Gensim [27] on literary books in the BCCWJ corpus [28]. We then extracted embeddings of fictional characters from J-KAC using quotation-annotated text with a speaker window of size one based on [19], which means speakers of one preceding turn and one succeeding turn were used. Note that the embeddings were obtained individually among different books. Therefore, we could obtain the embeddings of unseen characters who did not appear in the training data set. The dimension of fictional character embedding was 256.

The VQ module had a codebook size of 64 and code dimension of 256. We used HiFi-GAN [29] with officially distributed UNIVERSAL_V1 parameters as a vocoder.

For optimization, we used Adam [30]($\beta_1 = 0.9, \beta_2 = 0.99$) with learning-rate scheduling in a similar manner to a previous study [3]. The number of warm-up steps for the scheduling was 4000. The batch size was 4 and gradients were accumulated every 2 steps. For L_{TTS}, we used the sum of the L1 distances of pitch, energy, duration, and the mean square error of the mel-spectrogram. We empirically set the hyperparameters for the proposed model training as $\lambda_1 = 0.1$, $\lambda_2 = 0.1$, and $\beta = 0.1$.

During inference, we introduced the temperature softmax function to the attention layer of character-acting-style-extraction module with $T = 2$ so that the synthetic speech could be more expressive.

3.2. Evaluation Methods

3.2.1. Naturalness of synthetic speech

To evaluate the naturalness of synthetic speech, we conducted five-scale mean opinion score (MOS) tests. To assess the naturalness not only in isolated utterances but also in surrounding contexts, the MOS tests were conducted at both chapter and sentence levels. The chapter-level speech sample consisted of two to nine utterances. To make chapter-level speech, we concatenated utterances with a 400-ms silence. For the MOS test at the chapter level, the number of raters was 120 and each rater evaluated 12 samples in total. For the MOS test at the sentence

[4]https://github.com/Wataru-Nakata/FastSpeech2-JSUT

[5]https://huggingface.co/rinna/japanese-roberta-base

Table 2: *The results from Naturalness MOS test. ± indicates 95% confidence intervals.*

Model	Sentence level	Chapter level
FS2 (w/o BERT)	3.25 ± 0.13	3.01 ± 0.12
FS2	3.27 ± 0.13	3.02 ± 0.12
FS2-ResCNN	3.21 ± 0.14	3.03 ± 0.13
FS2-ResCNN-VQ	3.41 ± 0.12	3.35 ± 0.12
FS2-character	2.69 ± 0.14	2.46 ± 0.13
FS2-all	3.02 ± 0.12	2.97 ± 0.12

Table 3: *Results from AB test regarding effect of different modules in terms of character distinction. Results with statistical significance are shown in **bold text**.*

Method A	Score	Method B	p-value
FS2 (w/o BERT)	**0.44 vs. 0.56**	FS2	0.011
FS2	**0.32 vs. 0.68**	FS2-ResCNN	< 0.01
FS2-ResCNN	0.51 vs. 0.48	FS2-ResCNN-VQ	0.58

level, the number of raters was 60 and each rater evaluated 24 samples in total.

3.2.2. *Effect of different modules in terms of character distinction*

To investigate how the ResCNN and VQVAE in FS2-all affected character distinction, we conducted AB tests using the samples from FS2 (w/o BERT), FS2, FS2-ResCNN, and FS2-ResCNN-VQ. In these tests, raters were instructed to "listen to the two audiobook speech samples and choose the one that is more appropriate for character distinction on the basis of script below.". The number of raters was 60 and each rater evaluated 12 samples.

3.2.3. *Character acting style prediction*

To examine how well FS2-all can predict appropriate character-acting style, we conducted AB tests using samples from it as well as FS2-character and FS2-ResCNN-VQ. The same instruction was presented to raters as the investigation on modules above. The number of raters was 60 and each rater evaluated 12 samples.

4. Results

Table 2 shows the results from the naturalness MOS test using sentence-level and chapter-level samples. The naturalness of synthetic speech by FS2-character was significantly lower than those with the other models at both chapter and sentence levels. For FS2-all, while the naturalness at the chapter level was comparable to those with FS2 (w/o BERT) and FS2, it was significantly worse than that of FS2-ResCNN-VQ ($p < 0.01$). The MOS scores were degraded from sentence level to chapter level except with FS2-ResCNN-VQ. This result suggests that the prediction of suitable character acting styles can be a crucial, especially at the chapter level. At the chapter level, the MOS of FS2-ResCNN-VQ was higher than that of FS2-ResCNN. This indicates that vector quantization effectively helps improve naturalness. A possible reason is that the discrete nature of the VQ-VAE guaranteed the naturalness of unseen speech style, which sometimes fails when VQ is not used.

Table 3 shows the results from the AB test regarding the effect of different modules in terms of character distinction. Conditioning the speech synthesis model with RoBERTa and ResCNN had a positive effect in terms of character distinction. The use of VQ did not always have a negative impact on character distinction, even though the discrete embedding space learned with the VQVAE may put different characters into the same cluster. This can be due to the successful training of VQ-VAE, i.e., the FS2-ResCNN-VQ learned an appropriate space of character-acting styles.

Table 4: *The result from AB test for character-acting-style prediction. Results with statistical significance are shown in **bold text**.*

Method A	Score	Method B	p-value
FS2-character	**0.39 vs. 0.61**	FS2-ResCNN-VQ	< 0.01
FS2-character	**0.43 vs. 0.57**	FS2-all	< 0.01
FS2-ResCNN-VQ	**0.62 vs. 0.38**	FS2-all	< 0.01

Table 5: *Results from supplementary AB test for character-acting-style prediction. Results with statistical significance are shown in **bold text**.*

Method A	Score	Method B	p-value
FS2 (w/o BERT)	**0.44 vs. 0.56**	FS2	0.011
FS2 (w/o BERT)	**0.41 vs. 0.59**	FS2-all	< 0.01
FS2	**0.40 vs. 0.60**	FS2-all	< 0.01

Table 4 shows the results for predicting character-acting-style. FS2-all performed significantly better in terms of character distinction compared with FS2-character. Since FS2-character and FS2-all have the same inputs, this result indicates that using the fictional character embedding to predict character acting style works better than simply conditioning the speech synthesis model with character embedding.

5. Discussion

In Section 4, we showed that FS2-all outperformed FS2-character in terms of character distinction. However, this result could be due to the lack of naturalness of speech samples from FS2-character, as shown in Table 2. To further support our argument, we conducted a supplementary AB test with the samples from FS2 (w/o BERT), FS2, and FS2-all in a similar manner to the AB test for character-acting-style prediction. The results are shown in Table 5. We can confirm that our proposed method FS2-all was better than both FS2 (w/o BERT) and FS2 which have similar naturalness to FS2-all. Thus, the prediction of character-acting style helps achieve character distinction in synthetic speech.

6. Conclusions

We proposed a speech synthesis model which is conditioned by the predicted character-acting style from the quotation-annotated text for audiobook speech synthesis. The subjective-evaluation results shows that our proposed method can significantly improve on character distinction while having comparable naturalness at the chapter level with other models that do not use ground truth speech as input. We also showed that the use of discrete features as character-acting styles helps improve naturalness.

Future work includes improving character-acting-style prediction and introducing explicit constraint for distinction of character acting style.

7. References

[1] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proc. AAAI 2019*, vol. 33, no. 01, 2019, pp. 6706–6713.

[2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions," in *Proc. ICASSP 2018*, 2018, pp. 4779–4783.

[3] 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*, 2021.

[4] R. J. Weiss, R. Skerry-Ryan, E. Battenberg, S. Mariooryad, and D. P. Kingma, "Wave-Tacotron: Spectrogram-free end-to-end text-to-speech synthesis," in *Proc. ICASSP 2021*, 2021, pp. 5679–5683.

[5] K. Simon, C. Jane, M. Amy, and W. Lovisa, "The blizzard challenge 2018," in *Proc. Blizzard Challenge workshop*, 2018.

[6] E. Szekely, T. Csapó, B. Gyires-Tóth, P. Mihajlik, and J. Carson-Berndsen, "Synthesizing expressive speech from amateur audiobook recordings," in *Proc. IEEE SLT 2012*, 2012, pp. 297–302.

[7] S. Jung and H. Kim, "Pitchtron: Towards audiobook generation from ordinary people's voices," in *arXiv preprint*, 2020.

[8] W. Nakata, T. Koriyama, S. Takamichi, N. Tanji, Y. Ijima, R. Masumura, and H. Saruwatari, "Audiobook speech synthesis conditioned by cross-sentence context-aware word embeddings," in *Proc. 11th ISCA SSW*, 2021, pp. 211–215.

[9] G. Xu, W. Song, Z. Zhang, C. Zhang, X. He, and B. Zhou, "Improving prosody modelling with cross-utterance BERT embeddings for end-to-end speech synthesis," in *Proc. ICASSP 2021*, 2021, pp. 6079–6083.

[10] J. Pan, L. Wu, X. Yin, P. Wu, C. Xu, and Z. Ma, "A chapter-wise understanding system for text-to-speech in chinese novels," in *Proc. ICASSP 2021*, 2021, pp. 6069–6073.

[11] D. Stanton, Y. Wang, and R. J. Skerry-Ryan, "Predicting Expressive Speaking Style from Text in End-To-End Speech Synthesis," in *Proc. IEEE SLT 2018*, 2018.

[12] P. Wu, Z. Ling, L. Liu, Y. Jiang, H. Wu, and L. Dai, "End-to-end emotional speech synthesis using style tokens and semi-supervised training," in *Proc. APSIPA ASC 2019*, 2019, pp. 623–627.

[13] T. Li, S. Yang, L. Xue, and L. Xie, "Controllable emotion transfer for end-to-end speech synthesis," in *Proc. ISCSLP*, 2021, pp. 1–5.

[14] S. Kato, Y. Yasuda, X. Wang, E. Cooper, S. Takaki, and J. Yamagishi, "Modeling of rakugo speech and its limitations: Toward speech synthesis that entertains audiences," *IEEE Access*, vol. 8, pp. 138 149–138 161, 2020.

[15] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, 2018.

[16] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *Proc. ICASSP 2018*, 2018, pp. 5274–5278.

[17] E. Greene, T. Mishra, P. Haffner, and A. Conkie, "Predicting character-appropriate voices for a TTS-based storyteller system," in *Proc. INTERSPEECH 2012*, 2012, pp. 2207–2210.

[18] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," in *Proc. NIPS*, vol. 30, 2017.

[19] M. Azab, N. Kojima, J. Deng, and R. Mihalcea, "Representing movie characters in dialogues," in *Proceedings of the 23rd CoNLL*. Hong Kong, China: Association for Computational Linguistics, 2019, pp. 99–109.

[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, and P. G. Allen, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," in *arXiv preprint*, 2019.

[21] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep Speaker: an end-to-end neural speaker embedding system," *arXiv preprint*, vol. abs/1705.02304, 2017.

[22] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with Tacotron," in *Proceedings of the 35th International Conference on Machine Learning*, vol. 80, 2018, pp. 4693–4702.

[23] A. Lee and T. Kawahara, "Recent development of open-source speech recognition engine Julius," in *Proc. APSIPA ASC 2009*, 2009, pp. 131–137.

[24] 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.

[25] A. Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *Proc. ICASSP 2021*, 2021, pp. 6588–6592.

[26] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proc. ACL 2020*, 2020, pp. 8440–8451.

[27] R. Řehůřek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010, pp. 45–50.

[28] K. Maekawa, M. Yamazaki, T. Ogiso, T. Maruyama, H. Ogura, W. Kashino, H. Koiso, M. Yamaguchi, M. Tanaka, and Y. Den, "Balanced corpus of contemporary written Japanese," *Lang Resources & Evaluation*, vol. 48, pp. 345—-371, 2014.

[29] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Porc. NIPS 2020*, 2020.

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