High-Quality Foley Sound Synthesis using Monte Carlo Dropout

Technical Report

Chae-Woon Bang

Chosun University
Gwangju 61452, South Korea
bcw4045@chosun.kr

Nam Kyun Kim

Korea Automotive Technology Institute
Gwanjgu 62465, South Korea
kimnk@katech.re.kr

Chanjun Chun[*]

Chosun University
Gwangju 61452, South Korea
cjchun@chosun.ac.kr

ABSTRACT

This technical report describes the foley sound synthesis system for DCASE2023 Task7. Here, it aims to creates foley sound, which is widely utilized as various sound effects in multimedia contents. To accomplish this, it uses sound synthesis technique, generating a 4-second audio clip of one of seven classes. Specifically, we fine-tuned the baseline model such that improves the performance. After that, we ensemble the models using Monte Carlo Dropout. The performance of the proposed system was compared with the baseline using Frechet Audio Distance(FAD), which is referred as an audio evaluation metric. As a result, it was confirmed that both the single model and the ensemble model outperform the existing baseline system.

Index Terms— foley sound synthesis, Monte Carlo Dropout, model ensemble

1. INTRODUCTION

Foley sound refers to sound effects generated by events occurring in radio or movies. This foley sound is employed to add various sound effects in multimedia contents. The conventional foley sound synthesis was manually recorded and mixed by foley artists. However, recently, with advancements in generative models, research is being conducted to utilize sound synthesis techniques to generate foley sounds [1].

DCASE Task7 is to utilize sound synthesis technology to generate foley sounds. It consists of seven classes and generates sounds of 4 seconds in length.

In this technical report, we propose suitable hyperparameters based on the baseline provided by DCASE to perform high-quality sound generation. Furthermore, we suggest an ensemble system using Monte Carlo Dropout [2]. In other words, we fine-tuned the baseline model very sensitively to improve model behavior. Moreover, we utilized the Monte Carlo Dropout technique to facilitate ensemble training for models with long training times.

This technical report is structured as follows. Section 2 describes the model structure and training method of the proposed system. Section 3 describes the performance comparison of the baseline provided by DCASE and our proposed model. Finally, Section 4 describes the conclusion of this technical report.

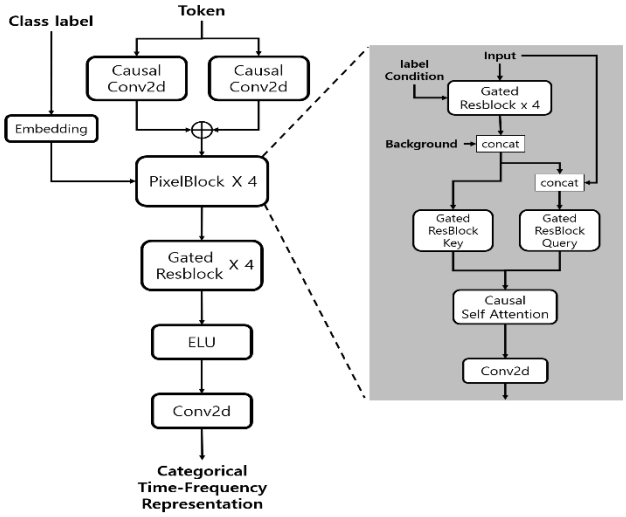

Figure 1: Overall architecture of PixelSNAIL

2. PROPOSED METHED

2.1. Dataset

The data provided by DCASE consists of a total of 4,850 sounds divided into seven classes (*DogBark*, *Footstep*, *Gunshot*, *Keyboard*, *MovingMotorVehicle*, *Rain*, *Sneeze/Cough*). The dataset was collected from UrbanSound8K, FSD50K, and BBC Sound Effects1, and seven classes were selected considering urban sound taxonomy [3]. All audio files have been converted to mono 16-bit format and have a sampling rate of 22,050Hz. In addition, the length of each sound is four seconds, and the number of samples

Table 1: Test results for each model: using a validation dataset

Class Label	Frechet Audio Distance(FAD)				
	Baseline Model	Fine-Tuned Model	Ensemble Model (N=2)	Ensemble Model (N=5)	Ensemble Model (N=10)
DogBark	14.826	12.066	10.928	11.085	11.450
Footstep	7.590	8.801	7.429	8.548	7.673
Gunshot	9.257	7.487	8.971	7.184	7.580
Keyboard	7.409	5.396	5.406	5.644	5.452
Moving MotorVehicle	17.605	15.848	17.035	16.063	16.036
Rain	15.130	14.211	11.985	12.819	14.534
Sneeze/Cough	2.840	2.266	2.743	2.782	3.03
Average	10.665	9.439	9.213	9.160	9.394

in each class is different. In this study, a window size of 1024 and a hop length of 256 were set to extract 80-dimensional mel-spectrograms, respectively. This is similar to the baseline.

2.2. Model architecture

The baseline system consists of a total of 3 modules: One of them includes PixelSNAIL, which is known as a generative model that combines causal convolution and self-attention mechanism to generate high-quality distributions [4]-[6]. PixelSNAIL takes the class label of the sound to be generated as input and generated a discrete time-frequency representation(DTFR). The following module is a VQ-VAE model that performs effective representation learning through vector quantization [7]. We use a trained VQ-VAE to acquire the DTFR generated from a mel-spectrogram [6]. In order to reconstruct a time-domain audio signal from the log mel-spectrogram, The HiFi-GAN, which is widely known as a high-performance neural Vocoder, was utilized [8].

Figure 1 shows the model structure of PixelSNAIL [6]. Here, The token is a vector filled with all zeros in the same shape as the DTFR in VQ-VAE. The PixelBlock consists of a combination of gated residual block and causal self-attention mechanism. The gated residual block regulates the information flow between layers using a gated activation unit and residual block [6]. It allows for controlled communication of information between different layers. The causal self-attention is utilized to extract crucial information by capturing the relationships and dependencies among the elements of the data. In other words, it helps capture important dependencies and patterns in the data by considering the causal relationships within the sequence [9].

The convolutional encoder used in VQ-VAE captures context-related acoustic information at various scales through a multi-scale convolution layer and then converts a discrete T-F representation(DTFR). The convolutional decoder is responsible for reconstructing the extracted time-frequency representation into a mel-spectrogram. The structure of the decoder uses a structure similar to that of the encoder, but the only difference is that it does not use a multi-scale convolution layer [7].

2.3. Training method

Adam was used as an optimizer for model learning. By setting the Cycle Scheduler, the learning rate is increased up to 3e-4, and the learning rate is adjusted periodically through the cosine period function [10]. Moreover, PixelSNAIL used 4 PixelBlocks, and Dropout was set to 0.1. In order to an appropriate model for PixelSNAIL, we performed several training with different channel information.

3. EXPERIMENTS

In order to demonstrate the effectiveness of our foley sound system, the objective evaluation was conducted using Frechet Audio Distance(FAD), which is widely employed as an audio evaluation metric [11]. Note that there was no overlap between the training and evaluation data. For each class, we utilized approximately 50 samples for evaluation.

Table 1 shows the FAD result for each model. The baseline model in Table 1 indicates 256-channel PixelBlock, while the fine-tuned model implies 512-channel PixelBlock. To ensemble several models, we utilized the Monte Carlo Dropout, where *N* represents the number of inferencing iterations. The experimental result indicates that our fine-tuned model and ensemble model outperform the baseline model in most classes. Among them, the ensemble model with 5 inferencing iterations showed the highest performance.

4. CONCLUSION

In this technical report, we attempted to fine-tune the baseline model for high quality sound synthesis, and ensembled the systems via Monte Carlo Dropout. The results of the evaluation through the validation dataset show that the single model and the ensemble model have high performance compared to the baseline model.

5. ACKNOWLEDGMENT

This work was supported by the “Science and Technology Project Opens the Future of the Region” program of Innopolis Foundation funded by Ministry of Science and ICT(2022-DD-UP-0312).

6. REFERENCES

[1] K. Choi, J. Im, L. Heller, B. McFee, K. Imoto, Y. Okamoto, M. Lagrange, and S. Takamichi, “Foley sound synthesis at the DCASE 2023 challenge,” *arXiv: 2304.12521*, Apr. 2023.

[2] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” in

Proc. International Conference on Machine Learning(*ICML*), pp. 1050–1059, June 2016.

[3] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. the 22nd ACM International Conference on Multimedia*, pp. 1041–1044, Nov. 2014.

[4] A. V. D. Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, and K. Kavukcuoglu, "Conditional image generation with PixelCNN decoders," in *Proc. Advances in Neural Information Processing Systems*(*NeurIPS*), pp. 4790–4798, June 2016.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, and A. N. Gomez, "Attention is all you need," in *Proc. Advances in Neural Information Processing Systems(NeurIPS)*, vol. 30, pp. 6000-6010, Dec. 2017.

[6] X. Chen, N. Mishra, M. Rohaninejad, and P. Abbeel, "PixelSNAIL: An improved autoregressive generative model," in *Proc. International Conference on Machine Learning (ICML)*, pp. 864–872, June 2018.

[7] A. V. D. Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Proc. Advances in Neural Information Processing Systems(NeurIPS)*, vol. 30, pp. 6306–6315, Dec. 2017.

[8] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Proc. Advances in Neural Information Processing Systems(NeurIPS)*, vol. 33, pp. 17022–17033, Oct. 2020.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, pp. 770-778, June 2016.

[10] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. IEEE Winter Conference on Applications of Computer Vision(WACV)*, pp. 464–472, Apr. 2017.

[11] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet audio distance: A metric for evaluating music enhancement algorithms," *arXiv: 1812.08466*, Dec. 2018.