Toward Leveraging Pre-Trained Self-Supervised Frontends for Automatic Singing Voice Understanding Tasks: Three Case Studies

Yuya Yamamoto
University of Tsukuba, Tsukuba, Japan
E-mail:s2130507@u.tsukuba.ac.jp

Abstract—**Automatic singing voice understanding tasks, such as singer identification, singing voice transcription, and singing technique classification, benefit from data-driven approaches that utilize deep learning techniques. These approaches work well even under the rich diversity of vocal and noisy samples owing to their representation ability. However, the limited availability of labeled data remains a significant obstacle to achieving satisfactory performance. In recent years, self-supervised learning models (SSL models) have been trained using large amounts of unlabeled data in the field of speech processing and music classification. By fine-tuning these models for the target tasks, comparable performance to conventional supervised learning can be achieved with limited training data. Therefore, in this paper, we investigate the effectiveness of SSL models for various singing voice recognition tasks. We report the results of experiments comparing SSL models for three different tasks (i.e., singer identification, singing voice transcription, and singing technique classification) as initial exploration and aim to discuss these findings. Experimental results show that each SSL model achieves comparable performance and sometimes outperforms compared to state-of-the-art methods on each task. We also conducted a layer-wise analysis to further understand the behavior of the SSL models.**

I. Introduction

The singing voice plays an important role in the music. It provides emotional expressions for us through its melody and lyrics. Computational understanding tasks of singing voices, such as singer identification, singing voice transcription, and singing expression identification, are beneficial for many applications such as music discovery [1], pedagogy [2], musicological analysis [3], etc. Processing the singing voices is a long-running challenge in the music information retrieval (MIR) field [4] due to its wide variation. Recently, the methods based on deep learning outperformed the conventional methods based on the hand-crafted manner in many singing voice understanding tasks thanks to its intense expressiveness [5], [6], [7], [8]. However, deep learning approaches typically necessitate extensive datasets comprising sung tracks with high-quality labels, entailing substantial costs for both data collection and annotation.

Transfer learning is one of the techniques to alleviate the requirements of large-scale datasets for the low-resource situation. It is based on the transfer of the knowledge derived from high-resource upstream pre-training tasks to target downstream tasks. In the audio domain, there are many works on audio classification tasks that utilize transfer learning of PANNs [9] and VGGish [10], which are pre-trained on a large-scale audio dataset. Notably, transfer learning of the model that is pre-trained by self-supervised learning (SSL) fashion is rapidly emerging. SSL models are leveraging a vast amount of unlabeled data, several notable models have been developed in both the speech and music domains. In the speech domain, these models include Wav2Vec2.0 [11], HuBERT [12], and WavLM [13]. Meanwhile, in the music domain, models such as MERT [14], and MapMusic2Vec [15] have been introduced.

Singing voice encompasses characteristics of both speech and music, and researchers have explored leveraging pre-trained SSL models in singing voice understanding tasks [16], [17], [18], [19] for each. The transfer learning of pre-trained SSL models from the speech or music domains to the singing domain holds potential. However, there is still ample room for investigating the utility of each pre-trained SSL model. This includes investigating which domains can contribute to singing voice understanding tasks, how the model extracts valuable features for the target tasks, and how to fully utilize the potential of the SSL models, and so on.

In this study, our objective is to examine the usefulness of SSL models pre-trained on the speech or music domains for singing voice understanding tasks. To achieve this, we employ pre-trained SSL models as a front-end to our voice identification model and evaluate their performance through fine-tuning.

We present the following contributions in this study:

1) Comparative analysis of SSL models: We compare multiple SSL models across three distinct three tasks: singer identification, singing voice transcription, and singing technique classification, corresponding to ”Who sings?”, ”What song?”, and ”How to sing?”, respectively.
2) Comparison with SoTA models: We also compared the SSL models with state-of-the-art (SoTA) models. Through fine-tuning the pre-trained SSL models, we demonstrate that they achieve performance comparable to that of current state-of-the-art methods in several singing voice understanding tasks.
3) Investigation of layer-wise behavior: Additionally, we delve deeper into the behavior of the model's layers and analyze their characteristics across different tasks,

by utilizing learnable weight for each layer.

II. Related works

Various machine learning approaches have been explored for low-resource problems in the singing voice understanding tasks. Semi-supervised learning [20], [21], data augmentation [22], [23], and self-supervised learning on singing voices [24], [25], leveraging speech data [26], [27] etc. have been proposed to mitigate the drawbacks of supervised-learning fashion.

More recent, pre-trained self-supervised models are used for singing understanding tasks. Ou et al. leveraged Wav2Vec 2.0 [11] model that is pre-trained using Librispeech corpus [28] and fine-tuned automatic speech recognition for lyric transcription. They achieved comparable performance with the state-of-the-art performance of lyric transcription methods [29], [30], which are learned from over 150 hours of data, by using its 10% of amount (i.e., 15 hours) [16]. Gu et al. leveraged Wav2Vec2.0 for singing voice transcription and outperformed conventional works [18]. Heydari et al. tackled singing beat tracking, which is a beat tracking method when the input is only a singing voice. They leverage WavLM [13] and DistilHuBERT [31] for the frontend of the model, and they outperformed the model that adapts only the spectrogram for the frontend feature. Donahue et al. proposed melody transcription from the musical mixture utilizing codified music representation [32] derived from the hidden representation of JukeBox [33], which is originally proposed for musical audio generation.

III. Methods

We compare four SSL models: 1) Wav2Vec2.0 [11], 2) WavLM [13], 3) MERT [14], and 4) MapMusic2Vec [15]. Each model takes a raw waveform as input and employs convolutional layers for feature extraction along with 12 Transformer encoder layers. The output of each model consists of a 768-dimensional vector per frame. We utilized these models as the frontend for each singing voice understanding task.

A. SSL models

1) Wav2Vec2.0: Wav2Vec2.0 [11] is the model that has convolutional and Transformer Encoder layers and acquired speech representation through contrastive learning and masking. It takes raw speech waveforms as input, and the initial convolutional layers produce latent representations denoted as z. To facilitate contrastive learning, the quantization module is applied to convert z into discrete representations denoted as Q. Simultaneously, z is fed into the Transformer layers after applying random masking to several frames. The output feature C is then derived from the Transformer layers. Finally, contrast learning is performed masked time step in C and Q. Here, the same time steps are considered as positive examples and different time steps are considered as negative examples. Wav2Vec2.0 has shown its effectiveness in several downstream speech tasks by fine-tuning [34], [35], [36]. We used the Wav2Vec2.0 Base model[1].

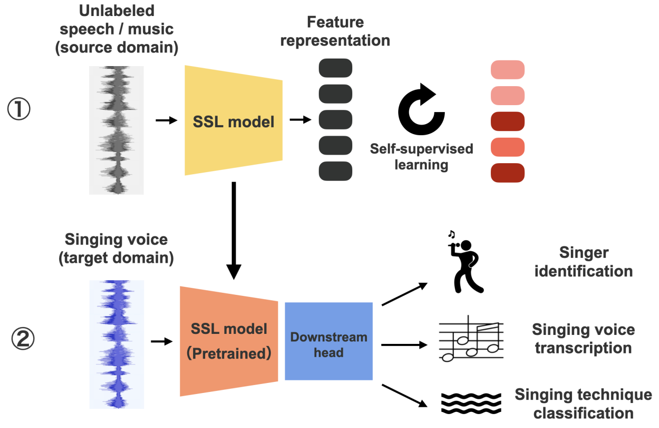
Unlabeled speech / music (source domain)
Feature representation
①
SSL model
Self-supervised learning
Singing voice (target domain)
②
SSL model (Pretrained)
Downstream head
Singer identification
Singing voice transcription
Singing technique classification

Fig. 1. The concept of this paper. 1) Pre-training on upstream task: Utilizing a vast amount of unlabeled data (either speech or music) and pre-training the model in a self-supervised fashion. 2) Transfer learning for downstream task: Leveraging pre-trained model and solving the target task (i.e., singing voice understanding tasks).

2) WavLM: WavLM [13] is a large-scale pre-trained model with 94k hours of speech data as input that can treat full-stack speech processing. It adopts masked prediction of hidden units like HuBERT [12] and denoising of the input speech as self-supervised pretraining. The diversity of the pre-trained corpus of WavLM is wider than that of Wav2Vec2.0; Gigaspeech [37], a collection of audiobooks, podcasts, and YouTube and VoxPopli[38], a collection of European Parliament, in addition to Librispeech. We used the Wav2Vec2.0 Base plus model, which demonstrates better performance than the Base model[2].

3) MERT: MERT [14] is a large-scale pre-trained model using unlabeled music data. It is also inspired by masked prediction of hidden units as WavLM while it uses CQT spectrogram as its target in addition to quantized acoustic features with the purpose of enhancing the pitch representative power. It can achieve the parameter efficiency compared to the JukeMIR [32], the musical audio representation derived from JukeBox. We used the public-v0 model[3], which is only trained on a public music dataset.

4) MapMusic2Vec: MapMusic2Vec[4] [15] is a model that is pre-trained using BYOL [39] with 1k hours of music data. It relies on two neural networks that have the same structure as each other, reffed to the teacher model and the student model. The parameters of the teacher model are updated according to the exponential moving average (EMA) of the student model. The student model takes partially masked audio raw waveform as input while the teacher model takes unmasked one and outputs the prediction of hidden outputs from the last K layers of the teacher model.

[1] https://huggingface.co/facebook/wav2vec2-base-960h

[2] https://huggingface.co/microsoft/wavlm-base-plus

[3] https://huggingface.co/m-a-p/MERT-v0-public

[4] https://huggingface.co/m-a-p/music2vec-v1

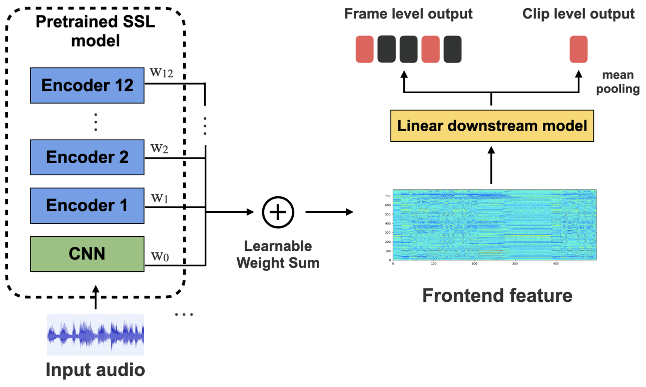

Fig. 2. The overview of the whole model to solve the target tasks.

B. Downstream model

We describe the overview of how to use the SSL model and the downstream models in Figure 2.

1) Weighted sum: We employed a weighted sum of the outputs from each Transformer encoder layer, including the input of the first layer, as the input for the downstream model. This approach was motivated by previous works (e.g.,[40], [41], [42], [43]) that have demonstrated different aspects of the input being captured by early, intermediate, and late layers of Transformer models. By using a weighted sum, we aimed to fully leverage the potential of SSL models. We set 13 learnable weight values, each corresponding to the weight value assigned to the output of each layer.

2) Classifier: Given the aforementioned features, we use a linear downstream model to predict each. The way of deriving the output of the target is depending on the tasks; For singing voice transcription, we directly used the frame-wise output. For the classification tasks (i.e., singer identification and singing technique classification), the feature is mean-pooled over the time axis to derive a clip-wise output.

C. Fine-tuning

In order to solve the downstream tasks, we fine-tune the models. There are various strategies for fine-tuning large-scale pre-trained models, we follow the two-stage training as Gu et al. [18] did. First, we freeze the parameter of the SSL models and make them learnable only on downstream models (i.e., the value of the weighted sum and the linear model). After several epochs, we unfreeze the Transformer encoders and fine-tune them.

IV. Experiments and Results

A. Experiment on Singer Identification

Singer identification is the classification task that identifies who is singing in a given sung audio clip.

1) Experimental Condition: We demonstrate 20-way singer identification using Artist20 [44] dataset, which collects 20 singers' music tracks. The dataset contains six albums for each singer in the data set, for a total of 1,413 songs. We split the dataset per album in order to avoid the leakage of production information about an album over the training and test set. We assign four albums for *train*, one album for *validation*, and the rest one album for *test* subset. Since the audio clips of the dataset include accompaniment of musical instruments, we applied vocal separation using Demucs V4 [45]. Then, we split them into five-second chunks without overlapping at a sample rate of 16kHz and discarded non-vocal chunks by RMS filtering [17].

For training the models, we used Adam optimizer [46] with 30 epochs. We set the learning rate of 3×10^{-3} for the first stage on the first six epochs and 5×10^{-5} for the second stage with the remaining epochs. The batch size is set to 32.

We evaluated the models by the following metrics: F1-score, Top-2 accuracy, and Top-3 accuracy. For the baseline for the comparison, we adopt the **CRNN** model by Hsieh et al.[23]. The model takes a 128-dimensional mel spectrogram as input and consists of four convolutional layers and two GRU [47] layers. We re-implemented the model in order to measure Top-2 and Top-3 accuracy since [23] only reported F1-score.

2) Results: Table I shows the results of singer identification. SSL models are demonstrating superior performance compared to the conventional SoTA model (i.e., CRNN) of singer identification. Notably, WavLM exhibited the highest performance in terms of F1 score, while MapMusic2Vec excelled in achieving the highest accuracy for Top-2 and Top-3 accuracy. These outcomes collectively suggest that the pre-training of SSL models with either music or speech data is leveraged in encoding the information associated with the singers.

TABLE I
The results of singer identification.

Methods	F1-score	Top-2	Top-3
Wav2Vec2.0	60.0	70.7	76.3
WavLM	**61.9**	70.2	76.4
MERT	56.8	68.4	75.6
MapMusic2Vec	59.6	**71.5**	**77.0**
CRNN [23]	49.5	63.4	71.3

B. Experiment on Singing Voice Transcription

Singing voice transcription refers to the process of converting sung audio signals into corresponding musical notes. In this paper, we employed the piano roll representation as the target for the singing voice transcription.

1) Problem Definition: We follow the settings of Wang et al. [48]. The target has four attributes: *onset*, *silence*, *pitch class*, and *octave*. The beginning of silence is considered as the offset instead of a direct estimation of them due to its difficulty. We set the pitch range from C2 (MIDI number 36, 65.41Hz) to B5 (MIDI number 83, 987.77Hz), therefore the target of *octave* is four classes (i.e., 2-5.) In addition, the class of inactive is added on *octave* and *pitch class*, respectively. Eventually, each frame contains 20-dimensional vectors as a target (i.e., *onset* and *silence* are binary, *pitch class* is five classes, and *octave* is 13 classes).

2) Experimental Condition: We use MIR-ST500 [48], which consists of 500 Chinese pop songs with manually annotated vocal melody notes. The authors of [48] provide the official data split that allocates 400 songs for the training set and 100 songs for the test set. Therefore, we followed the split as is, using the training set for the model training and the test set for the evaluation. Since the audio clips of the dataset include accompaniment of musical instruments, we applied vocal separation using Demucs V4 [45]. During the training of the model, we split the input audio into five-second chunks without overlaps at a sample rate of 16kHz. Given the input, the model outputs the prediction of the aforementioned 20-dimensional target vector per frame. The frame length is about 20ms. Suppose the prediction for *onset*, *silence*, *pitch class* and *octave* are $\hat{O}, \hat{S}, \hat{P}, \hat{V}$, the objective functions are as follows:

$$\mathcal{L}_{svt} = \frac{1}{T}\sum_{t=1}^{T}\Big[BCE(\sigma(\hat{O}_t), O_t, w_o) + BCE(\sigma(\hat{S}_t), S_t, w_s) + CE(\hat{P}_t, P_t) + CE(\hat{V}_t, V_t)\Big] \tag{1}$$

Where T denotes the number of frames in the input, $\sigma(\cdot)$ denotes the sigmoid function, BCE denotes binary cross-entropy loss, and CE denotes cross-entropy loss. Considering the imbalance between positive and negative samples, the weight is applied to the binary cross entropy loss. The values are $w_o = 15.0$ for onset, and $w_s = 1.0$ for silence, respectively. For training the models, we used Adam optimizer [46] with 30 epochs. We set the learning rate of 3×10^{-3} for the first stage and 5×10^{-5} for the second stage.

We follow the strategy of postprocessing as Gu et al. [18] for deriving the actual estimation. Briefly, each of the note properties is determined as follows:

- onset: Setting a threshold value of 0.4. If the prediction value is higher than the threshold and the local maximum, the frame is set to onset.
- offset: $\arg\,\min(\hat{S}_t > 0.5)$ of the estimated silence sequence and after the onset time.
- pitch class and octave (i.e., midi number): Assign the mode of the estimated value between the onset and the offset time.

To evaluate the performance of our model, we adopted three evaluation metrics proposed in [49], namely F1-score of COn (correct onset), COnP (correct onset and pitch), and COnPOff (correct onset, offset, and pitch). We utilized the mir_eval library to calculate these metrics, using the default parameters: 50 cents for pitch tolerance, 50 ms for onset tolerance, and the larger value between 50 ms and 0.2 of the note duration for offset tolerance.

In order to establish baselines for comparison, we considered several conventional works:

1) **EfficientNet-b0**: This approach is based on utilizing the EfficientNet-b0 model [50], which was originally proposed as the baseline for the MIR-ST500 task [48].
2) **JDC**$_{note}$: This model was trained on pseudo-labeled data obtained through quantization of automatically detected vocal melody contours [51].
3) **Wav2Vec2-Large**: This model employed the Wav2Vec2.0-Large model for the frontend and feeds only the last layer's output to the downstream model [18].

3) Results: Table II shows the results of singing voice transcription. MERT achieved the best score on COn and COnP among the four SSL models, while MapMusic2Vec demonstrated the best score on COnPOff. We observed that the speech models (i.e., Wav2Vec2.0 and WavLM) exhibit lower COnP compared to the music models (i.e., MERT and MapMusic2Vec). It suggests that the music models have already acquired the representation related to musical notes while the speech models lack such incorporation due to the gap between speech and singing voice (i.e., the length of stable pitch region, musical expression, etc.). In terms of performance comparisons with conventional works, every SSL model showed comparable performance with [48] and [51]. In addition, MERT outperforms Wav2Vec2Large in the COnP with fewer parameters.

TABLE II
RESULTS OF SINGING VOICE TRANSCRIPTION. ALL VALUES ARE EXPRESSED IN %. THE BEST-PERFORMING CONDITION AMONG ALL CONDITIONS IS HIGHLIGHTED IN BOLD, AND THE BEST-PERFORMING CONDITION AMONG SSL MODELS IS UNDERLINED.

Methods	COnPOff	COnP	COn
Wav2vec2.0	44.8	67.0	76.3
WavLM	44.4	67.3	76.9
MERT	46.7	**71.6**	78.2
MapMusic2Vec	50.7	70.0	77.9
EfficientNet-b0 [48]	45.8	66.6	75.4
JDC$_{note}$ [51]	42.2	69.7	76.2
Wav2Vec2-Large [18]	**52.4**	70.7	**78.3**

C. Experiment on Singing Technique Classification

Singing technique classification is the task that identifies a singing technique that appeared in a given input audio clip.

1) Experimental condition: We used VocalSet [52], which is a publicly available dataset that annotated singing techniques. VocalSet contains singing voices by 20 different professional singers (9 female and 11 male), performing 17 different singing techniques in various contexts. We selected ten techniques ("belt," "breathy," "inhaled singing," "lip trill," "spoken excerpt," "straight tone," "trill," "trillo," "vibrato," and "vocal fry") by all singers for the classification. We used the officially provided data split: 15 singers for the training set and 5 singers for the test set. We trimmed the silence from the audio and split it into non-overlapping chunks of 3 seconds. Typically the sampling rate of the audio tracks of VocalSet is 44.1kHz, the audio is resampled to 16kHz. For training the models, we used Adam optimizer [46] with 20 epochs. We set the learning rate of 3×10^{-3} for the first stage on the first five epochs and 5×10^{-5} for the second stage with the remaining epochs. The batch size is set to 16.

TABLE III
SELECTED SAMPLES FROM VOCALSET.

Label name	Type of fluctuation	Samples #
straight	None	1241
belt	Timbre	423
breathy	Timbre	455
vocal fry	Timbre, Modulation	587
vibrato	Modulation	1034
trill	Modulation	323
trillo	Modulation	242
lip trill	Modulation	376
inhaled	Other	151
spoken	Other	73

The data distribution of VocalSet is imbalanced over the classes that affect the classification performance [53]. Therefore, we adopt the inverse frequency weight for the loss function. The typical weight value w_c for the class c is as follows:

$$w_c = \frac{1}{(n_c)^{\alpha}} \tag{2}$$

where n_c is the number of training samples in class c, and α is the smoothing factor, which controls the smoothing of the loss weights. Note that α = 0 corresponds to the value of 1 (i.e., no weighting) and α = 1 corresponds to a reciprocal number (i.e., weighting by the inverse class frequency). We set α = 0.2, which performed the best score in [53]. We evaluated the models by the following metrics: F1-score, Accuracy, Balanced accuracy, Top-2 accuracy, and Top-3 accuracy. We consider several conventional works as baselines.

1) **1DCNN**: This model utilizes a CNN architecture that directly takes raw waveform as inputs. It serves as the official baseline model of the Vocalset dataset [52].
2) **OblongCNN**: This model employs a CNN architecture that takes a multi-resolution spectrogram, consisting of stacked representations with three different time-frequency resolutions as input. Additionally, it incorporates four convolutional layers of varying shapes [8].
3) **D-CNN-cRT**: This model replaces the standard convolutional layers with Deformable convolution and employs Classifier Retraining (cRT) [54] for training with a focus on addressing class imbalance [53].

2) Results: Table IV shows the results of singing technique classification. MapMusic2Vec exhibits the best performance in four SSL models and comparable performance to other conventional approaches. It also achieved higher accuracy compared to the best-performing method among the conventional works, D-CNN-cRT.

D. Layer-wise contribution analysis

We further analyze the weight of each encoder layer's output for each SSL model. Figure 3, 4, and 5 show the weights after training on singer identification, singing voice transcription, and singing technique classification, respectively. In each figure, 'Ln' denotes the weight of the n-th encoder layer's output (i.e., L0 is for the input of the first layer.).

TABLE IV
THE RESULTS OF SINGING TECHNIQUE CLASSIFICATION.

Methods	F1	Acc	BAcc	Top-2	Top-3
Wav2Vec2.0	56.1	58.3	60.1	74.8	82.1
WavLM	55.6	60.8	57.9	75.3	83.8
MERT	54.1	58.5	58.5	76.1	85.3
MapMusic2Vec	60.8	**66.0**	62.1	79.0	86.9
1DCNN [52]	48.8	58.4	48.4	76.4	86.3
OblongCNN [8]	51.3	55.4	57.5	74.3	85.8
D-CNN-cRT [53]	**62.0**	65.6	**65.5**	**81.5**	**88.7**

For singer identification, the strong contribution lies in the early layers of each SSL model. This finding aligns with previous research investigating the layer-wise contribution in various speech-related tasks, such as those presented in several works [13], [55]. These studies suggest that speaker-related information tends to be captured in the early layers of the models. Consequently, it can be inferred that a similar pattern holds true for singers, indicating that the crucial information for singer identification is also encoded in the early layers of the SSL models.

For singing voice transcription, except for MERT, the contribution also tends to lie in the early layers of each SSL model. According to [41], low-level prosodic features such as pitch or loudness tend to locate the early layers of speech SSL models. Since pitch information is the most important for singing voice transcription, the results that align with the work are unsurprising. In the case of MERT, CQT spectrogram is utilized as a prediction target. Therefore, it is plausible that the last layers of MERT contain pitch-related information. As several works [56], [57] reported, singing voice transcription is relating the information of phonemes as well as the pitch (i.e., onsets are often accompanied by the transition of the lyrics). It might cause this "unsmooth-shaped" distribution compared to other tasks.

In the case of singing technique classification, the early layers exhibit greater values compared to other layers in each model. Specifically, the first layer holds the greatest significance. It might be because singing techniques are influenced by pitch modulation and timbre variation. However, there is also a possibility that it is affected by the data imbalance where vibrato and straight, whose characteristics are the status of pitch modulation, are the majority classes.

V. CONCLUSION

In this study, we address the challenge of limited data availability in singing voice understanding tasks by leveraging transfer learning of pre-trained self-supervised (SSL) models. We employ four different SSL models as the frontend of our target task model. The models are subjected to comprehensive experiments encompassing singer identification, singing voice transcription, and singing technique classification. Our experimental results demonstrate that each SSL model achieves comparable and in some cases superior, performance when compared to conventional state-of-the-art models. Additionally, we conduct layer-wise analysis to inspect the behaviors, specif-

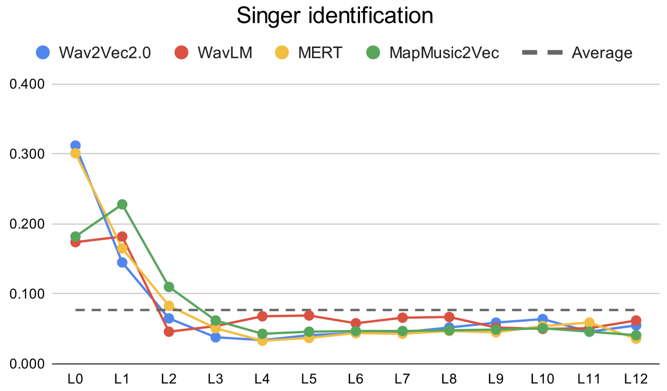

Fig. 3. The weights after training on singer identification.

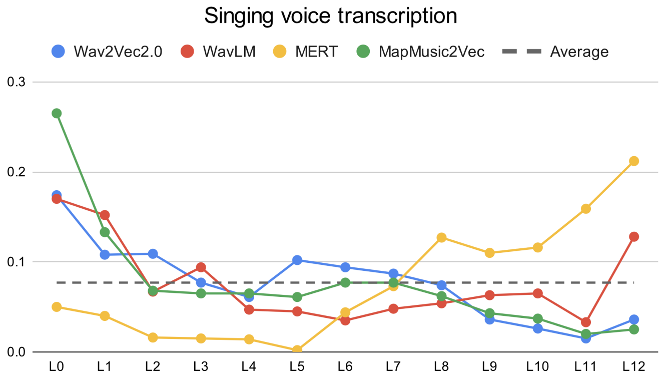

Fig. 4. The weights after training on singing voice transcription.

ically analyzing the weights of each layer. Moving forward, our future research endeavors will focus on further investigating the impact of feature representation in addressing the data scarcity issue in automatic singing voice understanding tasks. Furthermore, we propose that future studies can explore other singing voice understanding tasks, such as vocal melody extraction [58], lyric transcription [59], and singer diarization [60], among others, to expand the scope of research in this domain.

VI. Acknowledgements

I am grateful to Hiroko Terasawa and Juhan Nam, for their fruitful comments on the methods and evaluation of models. I also appreciate Jun-You Wang from National Taiwan University, for assisting in the preparation of *MIR-ST500*.

This work was supported by JST SPRING, Grant Number JPMJSP2124.

References

[1] A. Ghias, J. Logan, D. Chamberlin, and B. C. Smith, "Query by humming: Musical information retrieval in an audio database," in *Proceedings of the third ACM international conference on Multimedia*, 1995, pp. 231–236.

[2] T. Nakano, M. Goto, and Y. Hiraga, "Mirusinger: A singing skill visualization interface using real-time feedback and music cd recordings as referential data," in *Ninth IEEE International Symposium on Multimedia Workshops (ISMW 2007)*. IEEE, 2007, pp. 75–76.

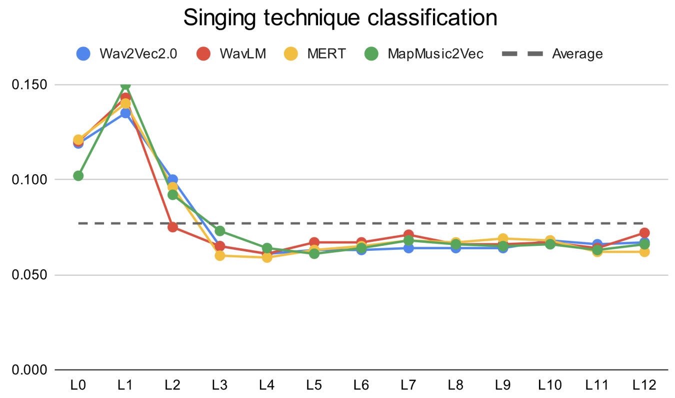

Fig. 5. The weights after training on singing technique classification.

[3] M. Panteli, R. Bittner, J. P. Bello, and S. Dixon, "Towards the characterization of singing styles in world music," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 636–640.

[4] E. J. Humphrey, S. Reddy, P. Seetharaman, A. Kumar, R. M. Bittner, A. Demetriou, S. Gulati, A. Jansson, T. Jehan, B. Lehner, A. Kruspe, and L. Yang, "An introduction to signal processing for singing-voice analysis: High notes in the effort to automate the understanding of vocals in music," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 82–94, 2018.

[5] C. Gupta, H. Li, and M. Goto, "Deep learning approaches in topics of singing information processing," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2422–2451, 2022.

[6] Z.-S. Fu and L. Su, "Hierarchical classification networks for singing voice segmentation and transcription," in *The 20th International Society for Music Information Retrieval Conference (ISMIR)*, 2019.

[7] Z. Nasrullah and Y. Zhao, "Music artist classification with convolutional recurrent neural networks," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[8] Y. Yamamoto, J. Nam, H. Terasawa, and Y. Hiraga, "Investigating time-frequency representations for audio feature extraction in singing technique classification," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 890–896.

[9] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[10] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2017, pp. 131–135.

[11] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in neural information processing systems*, vol. 33, pp. 12 449–12 460, 2020.

[12] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[13] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[14] Y. Li, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, C. Lin, A. Ragni, E. Benetos, N. Gyenge *et al.*, "Mert: Acoustic music understanding model with large-scale self-supervised training," *arXiv preprint arXiv:2306.00107*, 2023.

[15] Y. Li, R. Yuan, G. Zhang, Y. Ma, C. Lin, X. Chen, A. Ragni, H. Yin, Z. Hu, H. He *et al.*, "Map-music2vec: A simple and effective baseline

for self-supervised music audio representation learning," *arXiv preprint arXiv:2212.02508*, 2022.

[16] L. Ou, X. Gu, and Y. Wang, "Transfer learning of wav2vec 2.0 for automatic lyric transcription," in *The 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[17] M. Heydari and Z. Duan, "Singing beat tracking with self-supervised front-end and linear transformers," in *The 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[18] X. Gu, W. Zeng, J. Zhang, L. Ou, and Y. Wang, "Deep audio-visual singing voice transcription based on self-supervised learning models," *arXiv preprint arXiv:2304.12082*, 2023.

[19] C. Donahue, J. Thickstun, and P. Liang, "Melody transcription via generative pre-training," in *The 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[20] S. Kum, J.-H. Lin, L. Su, and J. Nam, "Semi-supervised learning using teacher-student models for vocal melody extraction," in *The 21st International Society for Music Information Retrieval Conference (ISMIR)*, 2020.

[21] J.-Y. Hsu and L. Su, "Vocano: A note transcription framework for singing voice in polyphonic music." in *The 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021, pp. 293–300.

[22] J. Schlüter and T. Grill, "Exploring data augmentation for improved singing voice detection with neural networks." in *The 16th International Society for Music Information Retrieval Conference (ISMIR)*, 2015, pp. 121–126.

[23] T.-H. Hsieh, K.-H. Cheng, Z.-C. Fan, Y.-C. Yang, and Y.-H. Yang, "Addressing the confounds of accompaniments in singer identification," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2020, pp. 1–5.

[24] H. Yakura, K. Watanabe, and M. Goto, "Self-supervised contrastive learning for singing voices," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1614–1623, 2022.

[25] C. Noufi and P. Verma, "Self-supervised learning of context-aware pitch prosody representations," *arXiv preprint arXiv:2007.09060*, 2020.

[26] S. Basak, S. Agarwal, S. Ganapathy, and N. Takahashi, "End-to-end lyrics recognition with voice to singing style transfer," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2021, pp. 266–270.

[27] C. Zhang, J. Yu, L. Chang, X. Tan, J. Chen, T. Qin, and K. Zhang, "Pdaugment: Data augmentation by pitch and duration adjustments for automatic lyrics transcription," in *The 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[28] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[29] E. Demirel, S. Ahlbäck, and S. Dixon, "Automatic lyrics transcription using dilated convolutional neural networks with self-attention," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–8.

[30] ——, "Mstre-net: Multistreaming acoustic modeling for automatic lyrics transcription," in *The 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021.

[31] H.-J. Chang, S.-w. Yang, and H.-y. Lee, "Distilhubert: Speech representation learning by layer-wise distillation of hidden-unit bert," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7087–7091.

[32] R. Castellon, C. Donahue, and P. Liang, "Codified audio language modeling learns useful representations for music information retrieval," in *The 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021.

[33] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, "Jukebox: A generative model for music," *arXiv preprint arXiv:2005.00341*, 2020.

[34] Y. Wang, A. Boumadane, and A. Heba, "A fine-tuned wav2vec 2.0/hubert benchmark for speech emotion recognition, speaker verification and spoken language understanding," *arXiv preprint arXiv:2111.02735*, 2021.

[35] N. Vaessen and D. A. Van Leeuwen, "Fine-tuning wav2vec2 for speaker recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7967–7971.

[36] C. Yi, J. Wang, N. Cheng, S. Zhou, and B. Xu, "Applying wav2vec2. 0 to speech recognition in various low-resource languages," *arXiv preprint arXiv:2012.12121*, 2020.

[37] G. Chen, S. Chai, G. Wang, J. Du, W.-Q. Zhang, C. Weng, D. Su, D. Povey, J. Trmal, J. Zhang *et al.*, "Gigaspeech: An evolving, multidomain asr corpus with 10,000 hours of transcribed audio," *arXiv preprint arXiv:2106.06909*, 2021.

[38] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," *arXiv preprint arXiv:2101.00390*, 2021.

[39] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar *et al.*, "Bootstrap your own latent-a new approach to self-supervised learning," *Advances in neural information processing systems*, vol. 33, pp. 21 271–21 284, 2020.

[40] H.-S. Choi, J. Lee, W. Kim, J. Lee, H. Heo, and K. Lee, "Neural analysis and synthesis: Reconstructing speech from self-supervised representations," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 251–16 265, 2021.

[41] G.-T. Lin, C.-L. Feng, W.-P. Huang, Y. Tseng, T.-H. Lin, C.-A. Li, H.-y. Lee, and N. G. Ward, "On the utility of self-supervised models for prosody-related tasks," in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 1104–1111.

[42] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.

[43] A. Pasad, J.-C. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," in *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2021, pp. 914–921.

[44] D. P. Ellis, "Classifying music audio with timbral and chroma features," in *The 8th International Conference for Music Information Retrieval Conference (ISMIR)*, 2007.

[45] S. Rouard, F. Massa, and A. Défossez, "Hybrid transformers for music source separation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2023.

[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[47] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[48] J.-Y. Wang and J.-S. R. Jang, "On the preparation and validation of a large-scale dataset of singing transcription," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2021, pp. 276–280.

[49] E. Molina, A. M. Barbancho-Perez, L. J. Tardon-Garcia, I. Barbancho-Perez *et al.*, "Evaluation framework for automatic singing transcription," in *The 15th International Society for Music Information Retrieval Conference (ISMIR)*, 2014.

[50] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[51] S. Kum, J. Lee, K. L. Kim, T. Kim, and J. Nam, "Pseudo-label transfer from frame-level to note-level in a teacher-student framework for singing transcription from polyphonic music," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2022, pp. 796–800.

[52] J. Wilkins, P. Seetharaman, A. Wahl, and B. A. Pardo, "Vocalset: A singing voice dataset," in *The Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, 2018, pp. 468–474.

[53] Y. Yamamoto, J. Nam, and H. Terasawa, "Deformable CNN and Imbalance-Aware Feature Learning for Singing Technique Classification," in *Proceedings of the 21st Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2022, pp. 2778–2782.

[54] B. Kang, S. Xie, M. Rohrbach, Z. Yan, A. Gordo, J. Feng, and Y. Kalantidis, "Decoupling representation and classifier for long-tailed recognition," in *International Conference on Learning Representations (ICLR)*, 2020.

[55] Z. Chen, S. Chen, Y. Wu, Y. Qian, C. Wang, S. Liu, Y. Qian, and M. Zeng, "Large-scale self-supervised speech representation learning for automatic speaker verification," in *Proceedings of the IEEE International*

Conference on Acoustics, Speech, and Signal Processing (ICASSP). IEEE, 2022, pp. 6147–6151.

[56] S. Yong, L. Su, and J. Nam, "A phoneme-informed neural network model for note-level singing transcription," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 2023.

[57] T. Deng, E. Nakamura, and K. Yoshii, "End-to-end lyrics transcription informed by pitch and onset estimation," in *The 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022.

[58] K. S. Rao, P. P. Das *et al.*, "Melody extraction from polyphonic music by deep learning approaches: A review," *arXiv preprint arXiv:2202.01078*, 2022.

[59] X. Gao, "Automatic lyrics transcription of polyphonic music," Ph.D. dissertation, National University of Singapore (Singapore), 2022.

[60] H. Suda, D. Saito, S. Fukayama, T. Nakano, and M. Goto, "Singer diarization for polyphonic music with unison singing," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 1531–1545, 2022.