Lead Instrument Detection from Multitrack Music

Longshen Ou
School of Computing
National University of Singapore
Singapore
oulongshen@u.nus.edu

Yu Takahashi
Research and Development Division
Yamaha Corporation
Hamamatsu, Japan
yu.takahashi@music.yamaha.com

Ye Wang
School of Computing
National University of Singapore
Singapore
wangye@comp.nus.edu.sg

***Abstract*—Prior approaches to lead instrument detection primarily analyze mixture audio, limited to coarse classifications and lacking generalization ability. This paper presents a novel approach to lead instrument detection in multitrack music audio by crafting expertly annotated datasets and designing a novel framework that integrates a self-supervised learning model with a track-wise, frame-level attention-based classifier. This attention mechanism dynamically extracts and aggregates track-specific features based on their auditory importance, enabling precise detection across varied instrument types and combinations. Enhanced by track classification and permutation augmentation, our model substantially outperforms existing SVM and CRNN models, showing robustness on unseen instruments and out-of-domain testing. We believe our exploration provides valuable insights for future research on audio content analysis in multitrack music settings.**

***Index Terms*—Audio content analysis, multitrack music, lead instrument detection, track-wise attention, feature fusion**

I. Introduction

Audio content analysis is a crucial area of study within audio processing and music information retrieval, focusing on tasks like identification, transcription, and segmentation [1]. Beyond these foundational tasks, a critical aspect of human musical perception involves identifying the **lead instrument**—the instrument that captures the listener's attention with its dominant auditory presence. This could range from the lead vocals [2] and guitar solos [3] in rock and pop, to lead saxophone or trumpet [4] as well as drum solos [5] in jazz. Automating lead instrument detection can not only facilitate the creation of audio thumbnails and music structural analysis but also potentially simplify audio mixing workflows and enhance music recommendation systems.

However, prior research on lead instrument detection has primarily focused on analyzing mixture audio [6]–[11] or isolated single instrument tracks [12]–[14], limiting their ability to capture high-level, instrument-specific properties like roles and interactions within a song, essential for identifying lead instruments. These studies often involve coarse-level classification, confined to predefined categories like vocals or guitar solos [11], [12], restricting applicability in real-world settings where any instrument can serve the lead role. Additionally, many works rely on Support Vector Machine (SVM) models [6]–[12], [15], constrained by the necessity to include all potential lead instruments in training data, thus failing to identify new instruments absent from training.

The analysis of multitrack music, where each track contains specific instrument audio in a time-synchronized, multi-stream format, remains significantly underexplored. To our knowledge, there is no existing work that outlines neural network designs specifically for handling multitrack audio inputs in content analysis tasks. While prior studies on automatic mixing tasks have utilized deep learning models [16]–[19], these approaches often assume a fixed number and type of instruments, which limits their applicability in real-world scenarios, where track counts and instrument types can vary significantly. Recently, [20] introduced a framework capable of handling arbitrary track combinations for automatic mixing, sharing some design similarities with our model.

The pre-train and fine-tune paradigm with self-supervised learning (SSL) models like wav2vec 2.0 [21] and HuBERT [22] has revolutionized audio and speech domains, advancing music information retrieval [23] and reducing reliance on in-domain data [24], [25]. Music-specific SSL models have also achieved state-of-the-art results in various tasks [26], but their use remains largely limited to single-stream audio. Extending these models to multitrack audio requires adapting single-track architectures to multitrack context and enabling effective cross-track feature integration. Addressing these issues is crucial for enhancing SSL applications in complex multitrack environments.

This paper develops neural network models for detecting the lead instrument at any moment in multitrack music audio, extending conventional vocal and guitar solo detection to any type of lead instrument while adapting to the multitrack setting without assumptions about track count or type. We created two expertly annotated datasets and compared various model designs. Our model uses a shared SSL audio encoder across tracks, with a novel track-wise attention mechanism that aggregates features from each instrument track based on its importance relative to the mixture track. To further enhance performance, we introduced a track permutation augmentation strategy to diversify the training data. Our key contributions include:

- **Initiate the task of lead instrument detection from multitrack music**, with expertly annotated datasets, and strong baseline models across multiple settings, including analyses for both segment-level and frame-level, both

multitrack and single mixture tracks.[1]

- **Superior performance compared to existing models**, including SVM- and CRNN-based approaches, and also generalizable to unseen instruments and new domains.
- **Evaluated multiple model designs**, demonstrating the advantages of the proposed track-wise attention and track classification.

II. Method

A. Problem Definition

The task involves identifying the lead instrument at any given timestep within a multitrack music recording, composed of time-synchronized audio tracks from different instruments of the same song. We assume only one lead instrument at each timestep, indicated by track number or instrument name. We also assume access to track-wise metadata including instrument type and track ID, as well as a human-produced mixture track, which we utilize to aid the detection.

B. Crafting Datasets

To establish a foundation for this new task, we created two datasets with expert annotations from two different audio sources. They include an internal dataset named MJN includes multitrack recordings from five live events, each featuring performances by 4 to 6 bands, and the MedleyDB dataset [27], known for its use in various MIR tasks. Annotations were performed by experts using Adobe Audition, who marked lead instruments based on audio playback and observing waveforms. The process involved identifying onsets, offsets, and instrument types, with specific rules for overlapping leads and minimum segment durations. For more details, please refer to our dataset documentation[1]. The resulting datasets were 7.10 h and 5.57 h in duration, with further elaboration on splitting strategies in Section III-A.

We highlight some important properties of the datasets. In both MJN and MedleyDB, frequencies of lead instrument type exhibit a long-tailed distribution, with vocals being the most frequent lead instrument, followed by electric guitar. Notably, the MJN dataset, commonly used open microphones on stage, contains considerable amount of bleeding sound. While MedleyDB offers a greater variety of instrument types—29 in total, with 26 as leads (14 and 13 in MJN), MJN shows a more balanced distribution among lead instruments, where the frequency of the second most common lead is 53% that of the most frequent, versus 37.1% in MedleyDB. Additionally, MJN features more frequent lead instrument switches (23.05 s per change) than MedleyDB (29.33 s per change).

C. Lead Instrument Detection Model

Our model, depicted in Figures 1 and 2, integrates an audio encoder with an attention-based classifier. Audio from each track is processed through a shared encoder to generate track-wise feature maps. We employ MERT [26], a music-specific self-supervised learning (SSL) model, as our audio encoder due to its robust performance in timbre and pitch-related tasks. Instrument types and track IDs are embedded and added to the feature maps: instrument embeddings reduce the encoder's burden by delegating timbre-to-instrument mapping, while track embeddings enables track classification. A track-wise attention mechanism aggregates these features into a single feature map, which a frame-wise linear classifier uses to produce the final classification results.

To aggregate multitrack information effectively, we designed a frame-level track-wise dot-product attention mechanism. For each frame, the mixture track serves as the query, while individual instrument tracks act as keys and values. This setup enables the dot product to perform a nuanced comparison of each instrument track against the global mixture, determining the importance level of each track at every timestep. Subsequently, features from all tracks are aggregated into a single feature map through a weighted sum, with weights derived from the initial comparisons' attention scores. This aggregation prioritizes tracks that contribute most significantly to the overall auditory effect. This design emphasizes track-to-mixture content comparisons to highlight the prominence (or relevance) of specific sound sources within a complex multitrack environment, mirroring the selective auditory attention of humans.

Direct instrument classification faces challenges with generalizability. It fails to accurately classify untrained instrument types, instead misclassifying them as similar-sounding instruments. Moreover, it cannot distinguish between multiple

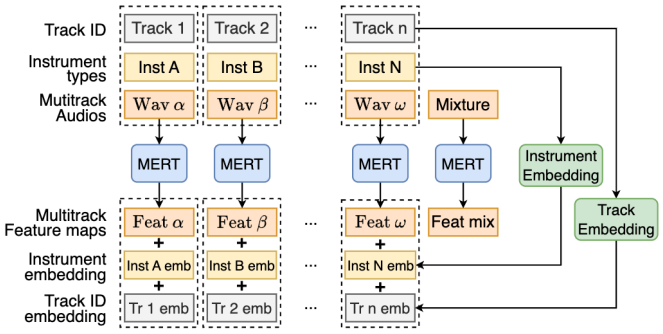

Fig. 1. Encoding information for each track.

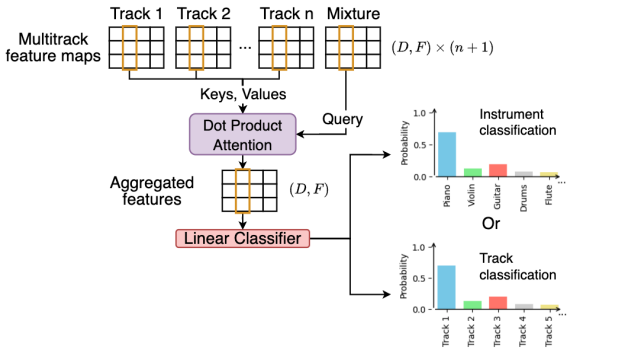

Fig. 2. Track-wise frame-level attention and subsequent classification.

[1]Please refer to https://github.com/Sonata165/LeadInstrumentDetection for code, dataset annotations, and model checkpoints.

instances of the same instrument type across different tracks, which limits its practical applicability. To overcome these issues, we adjusted the classification scheme from instrument types to track IDs of lead instruments. This modification not only enable generalization to unseen instruments but also improves performance in out-of-domain testing.

Despite the potential for higher generalization, track classification faces a challenge with the fixed content–track relationship, caused by consistent instrument type within a track throughout a performance, leading to homogenize feature map, biasing the attention-based classifier to make predictions based on simple track ID cues rather than actual audio content. We introduce track permutation augmentation to overcome this limitation. In training, track IDs are randomly permuted and reassigned across all tracks, with corresponding label adjustments. This approach effectively prevents the model from relying on track IDs for classification, thereby making the learning process more efficient and enhancing performance, while preserves the integrity of the multitrack data.

III. Experiments

A. Implementation Details

We primarily utilize the MJN dataset for our experiments because it is organized by performance and minimizes instrument changes, which simplifies the control of testing conditions. The validation set includes three challenging performances to test model robustness, while the test set features two typical band settings with a relatively balanced label distribution. The remaining 20 performances make up the training set. For MedleyDB, data is split at the song level, with 15% randomly allocated to both validation and test sets.

Audio are normalized to -0.1dB, converted to mono, resampled to 24000 Hz, and segmented into 5-second clips with 2.5-second overlaps. We use `MERT-v1-95M`[2] as our audio encoder with full parameter fine-tuning. The track-wise attention is implemented with a 12-head multihead attention, with layer normalization and large dropout (p=0.8) afterwards.

Training utilizes the AdamW optimizer [28] with a weight decay of 0.01. Learning rates are set at 1e-5 for both the audio encoder and track-wise attention, and 1e-3 for the linear classifier. Cross entropy loss is used as the objective function. The model undergoes two training epochs with a batch size of 4, supplemented by 4-step gradient accumulation to achieve a total batch size of 16. Validation occurs every quarter epoch, with checkpoint selection based on the Macro F1 score on the validation set. Training is performed on an RTX 3090 GPU (24GB).

B. Metrics

We utilize accuracy and Macro F1 score as metrics, calculated directly at the frame level. Each metric is first averaged over a 5-second sample and then across the entire test set. Accuracy measures the percentage of correct detections made by the model, while Macro F1 is chosen for its ability to equally weigh the performance of each class—crucial in our datasets, where lead instrument distribution is imbalanced. For instrument classification models, we calculate instrument F1 (**Inst F1**) and accuracy (**Inst Acc**) directly. For track classification models, we first compute Macro F1 for track predictions (**Track F1**), then map these to instrument types using the known track–instrument relationships to compute instrument-level Macro F1 and accuracy, facilitating cross-scheme comparisons.

[2]https://huggingface.co/m-a-p/MERT-v1-95M

TABLE I
Comparison of classification module designs. Cls. refers to classification schemes.

Model	Cls.	Validation set (hard)			Test set (balanced)		
		Track F1	Inst F1	Inst Acc	Track F1	Inst F1	Inst Acc
From mix	Inst.	-	66.78	73.88	-	57.17	72.69
Track avg.	Inst.	-	69.34	75.57	-	73.02	83.67
Track attn.	Inst.	-	77.42	83.96	-	83.32	**91.28**
Track attn.	Track	80.56	**83.66**	**87.79**	83.76	**83.76**	91.20

C. Baseline Models

To show the effectiveness of our track-wise attention, we implement two variants of our model: **From mix**, a straightforward implementation using a MERT model with a linear classifier processing only the mixture track; and **Track avg.**, where the classifier operates on the averaged feature maps from all tracks. Finally, **Track attn.** is our model incorporating track-wise attention.

Additionally, we compared against two external models. The first is a CRNN model from [29], designed for sound event detection tasks with multi-channel audio as inputs. The second baseline is an SVM model from [12], which focuses on segment-level binary classification of guitar solos using mixture audio. We adapt our "From mix" model for segment-level classification by implementing average pooling across all frames in the feature map before classification with a linear classifier.

IV. Results

A. Comparison on Classification Module Design

As Table I illustrates, the *track avg.* model aid performance with signal from instrument track to achieve better performance than the *from mix* model, but without enough robustness as the performance enhancement varies significantly between easier (+15.85% instrument F1 on the test set) and challenging cases (+2.56% on the validation set). Model with track-wise attention more effectively utilizes track information, as evidenced by gains in both validation (+8.08%) and test (+10.03%) sets, compared to *track avg.* model. Moreover, switching to track classification further enhances performance, particularly in difficult cases, with a notable +6.24% increase in instrument F1 on the validation set. Overall, the combination of track-wise attention and track classification gives the strongest performance.[3]

[3]Recordings in the test set lack multiple instances of the same instrument type within each performance, leading to identical Track F1 and Inst F1 values.

TABLE II
ABLATION STUDIES

Model	Validation set (hard)			Test set (balanced)		
	Track F1	Inst F1	Inst Acc	Track F1	Inst F1	Inst Acc
Ours	**80.56**	**83.66**	**87.79**	**83.76**	**83.76**	**91.20**
w/o track perm	63.71	69.55	70.20	78.44	78.44	89.59
w/o inst emb	68.15	71.60	74.29	78.88	78.88	89.36
w/o track emb	33.64	42.16	35.53	21.38	21.38	24.86
Freeze MERT	31.13	34.98	47.71	26.41	26.41	38.30
FT last layer	56.89	60.59	74.51	43.87	43.87	71.31
w/o oracle mix	80.48	82.94	86.88	83.08	83.08	90.44

TABLE III
CROSS-DATASET TESTING RESULTS

Cls	Training set		Test set (MJN)		Test set (MedleyDB)	
	MJN	MedleyDB	Inst F1	Inst Acc	Inst F1	Inst Acc
Inst.	Y		82.04	90.53	57.68	63.06
Track	Y		81.34	89.50	74.13	69.01
Track		Y	52.30	60.16	**84.72**	78.87
Track	Y	Y	**84.56**	**92.11**	84.47	**83.25**

Additionally, we demonstrate the generalization capability of track classification using the organ—an instrument not present in the training set, with timbre similar to that of the electric guitar. While instrument classification frequently mislabels it as electric guitar (75.68%), with no correct classifications, track classification achieves 62.42% accuracy, showcasing enhanced adaptability to unseen instruments.

B. Ablation Studies

Table II presents three sets of ablation study results. First, we assess the impact of omitting key components. The removal of track permutation significantly diminishes performance, particularly in challenging scenarios, and fails to achieve results comparable to the model with instrument classification in Table I. Similarly, excluding instrument embeddings leads to substantial performance losses. Eliminating track embeddings causes the track classification model to nearly fail at making predictions.

We then explore fine-tuning configurations to justify our choice of full parameter fine-tuning. Freezing the MERT model did not yield meaningful results, while fine-tuning its last transformer layer along with the attention and classifier significantly boosted performance, approaching that of our final model. This indicates that training more parameters in the audio encoder leads to higher performance.

Additionally, we evaluate the model's dependency on the human-produced mixture track by excluding it (*w/o oracle mix*) and replacing it with a pseudo mixture track, created by averaging all single-instrument tracks at the waveform level. This adjustment results in only a very slight performance decrease, suggesting that our model remains functional without the human-produced mixture track, making it applicable in fully automated scenarios.

C. Cross-Dataset Testing

Table III[4] presents the results for out-of-domain testing. When trained on the MJN dataset, the track classification model not only maintains strong in-domain performance but also excels in out-of-domain tests, showing a +16.45% improvement in instrument F1 on MedleyDB over the instrument classification model. Training with both datasets yields the best overall performance, highlighting the importance of data quantity and diversity. Moreover, training with MedleyDB and testing on MJN results in a significant performance drop (-32.26% in instrument F1), but this decline is less pronounced when reversing the training and testing sets (-10.34%), suggesting MJN is a more effective training set for this task. This observation highlight key strategies for future dataset construction with limited resources: tolerating data imperfections like bleeding sound and less instrument diversity, ensuring a more balanced distribution of lead instruments and more frequent switches of lead instruments.

[4]This experiment used a smaller batch size (=1), leading to slightly weaker performance on the MJN test set compared to Table I.

TABLE IV
COMPARISON WITH CRNN

Model	Input	FT MERT	Validation set (hard)		Test set (balanced)	
			Inst F1	Inst Acc	Inst F1	Inst Acc
Ours	MERT feat.	Y	**77.42**	**83.96**	**83.32**	**91.28**
CRNN	Mel spec.	NA	37.64	46.36	17.15	24.93
CRNN	MERT feat.	N	57.62	60.16	23.20	26.48
CRNN	MERT feat.	Y	57.62	60.16	23.20	26.48
CRNN + attn.	MERT feat.	Y	57.62	60.16	26.50	31.40

TABLE V
COMPARISON WITH SVM ON SEGMENT-LEVEL GUITAR SOLO DETECTION

Model	Validation set (hard)			Test set (balanced)		
	Acc	Guitar F1	Macro F1	Acc	Guitar F1	Macro F1
Ours	**93.02**	**82.17**	**88.91**	**92.20**	**87.50**	**90.91**
SVM	75.50	26.03	55.67	56.44	20.80	45.38

D. Comparison with Prior Works

1) CRNN: The comparison with CRNN is presented in Table IV. Overall, the CRNN model performs poorly on our task, regardless of whether the audio features used are mel spectrograms or MERT features, and whether MERT is fine-tuned or not. The performance gap between CRNN and our model is substantial.

2) SVM: As shown in Table V, the SVM model also struggles on our dataset, achieving only 26.03% F1 for guitar. In contrast, our model maintains competitive performance in segment-level classification, demonstrating a significant advantage over the SVM model.

V. CONCLUSION

In this paper, we introduced the task of lead instrument detection in multitrack music and developed two annotated datasets specifically for this purpose. Our proposed model, which incorporates an SSL audio encoder, instrument and track embeddings, track-wise attention, track classification, and track permutation augmentation, effectively addresses this task and is capable of generalizing to unseen instruments. We established robust baselines on these datasets and demonstrated the superiority of our approach through comparative studies with CRNN and SVM models. This exploration of multitrack audio content analysis provides valuable insights for future research on similar tasks.

REFERENCES

[1] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Transactions on speech and audio processing*, vol. 10, no. 7, pp. 504–516, 2002.

[2] M. Bürgel, L. Picinali, and K. Siedenburg, "Listening in the mix: Lead vocals robustly attract auditory attention in popular music," *Frontiers in Psychology*, vol. 12, p. 769663, 2021.

[3] B. Goertzel, "The rock guitar solo: From expression to simulation," *Popular Music & Society*, vol. 15, no. 1, pp. 91–101, 1991.

[4] J. Abeßer, E. Cano, K. Frieler, and M. Pfleiderer, "Dynamics in jazz improvisation. score-informed estimation and contextual analysis of tone intensities in trumpet and saxophone solos," in *Proceedings of the 8th Conference on Interdisciplinary Musicology (CIM14), Berlin*, 2014, pp. 4–6.

[5] Z. A. King, "A brief history of jazz drumming," 2014.

[6] A. Wieczorkowska, E. Kolczyńska, and Z. W. Raś, "Training of classifiers for the recognition of musical instrument dominating in the same-pitch mix," in *New Challenges in Applied Intelligence Technologies*. Springer, 2008, pp. 213–222.

[7] A. A. Wieczorkowska and E. Kubera, "Identification of a dominating instrument in polytimbral same-pitch mixes using svm classifiers with non-linear kernel," *Journal of Intelligent Information Systems*, vol. 34, pp. 275–303, 2010.

[8] G. Peterschmitt, E. Gomez, and P. Herrera, "Pitch-based solo location," in *Proc. of MOSART Workshop on Current Research Directions in Computer Music*, 2001.

[9] C. Smit and D. P. Ellis, "Solo voice detection via optimal cancellation," in *2007 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. IEEE, 2007, pp. 207–210.

[10] F. Fuhrmann, P. Herrera, and X. Serra, "Detecting solo phrases in music using spectral and pitch-related descriptors," *Journal of New Music Research*, vol. 38, no. 4, pp. 343–356, 2009.

[11] M. Mauch, H. Fujihara, K. Yoshii, and M. Goto, "Timbre and melody features for the recognition of vocal activity and instrumental solos in polyphonic music." in *ISMIR*, 2011, pp. 233–238.

[12] K. Pati and A. Lerch, "A dataset and method for electric guitar solo detection in rock music," in *Proc. AES Int. Conf. on Semantic Audio, Erlangen, Germany*, 2017.

[13] F. Pachet, P. Roy, J. Moreira, and M. d'Inverno, "Reflexive loopers for solo musical improvisation," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2013, pp. 2205–2208.

[14] R. Foulon, P. Roy, and F. Pachet, "Automatic classification of guitar playing modes," in *Sound, Music, and Motion: 10th International Symposium, CMMR 2013, Marseille, France, October 15-18, 2013. Revised Selected Papers 10*. Springer, 2014, pp. 58–71.

[15] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, "A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals." in *ISMIR*, 2012, pp. 559–564.

[16] M. A. Martínez Ramírez, D. Stoller, and D. Moffat, "A deep learning approach to intelligent drum mixing with the wave-u-net," *Journal of the Audio Engineering Society*, vol. 69, no. 3, pp. 142–151, 2021.

[17] C. J. Steinmetz, J. Pons, S. Pascual, and J. Serra, "Automatic multitrack mixing with a differentiable mixing console of neural audio effects," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 71–75.

[18] M. A. Martinez Ramirez, W. Liao, C. Nagashima, G. Fabbro, S. Uhlich, and Y. Mitsufuji, "Automatic music mixing with deep learning and out-of-domain data," in *Ismir 2022 Hybrid Conference*, 2022.

[19] J. Koo, M. A. Martínez-Ramírez, W.-H. Liao, S. Uhlich, K. Lee, and Y. Mitsufuji, "Music mixing style transfer: A contrastive learning approach to disentangle audio effects," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[20] S. S. Vanka, C. Steinmetz, J.-B. Rolland, J. Reiss, and G. Fazekas, "Diff-mst: Differentiable mixing style transfer," *arXiv preprint arXiv:2407.08889*, 2024.

[21] 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.

[22] 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.

[23] X. Gu, L. Ou, W. Zeng, J. Zhang, N. Wong, and Y. Wang, "Automatic lyric transcription and automatic music transcription from multimodal singing," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 20, no. 7, pp. 1–29, 2024.

[24] X. Gao, X. Yue, and H. Li, "Self-transriber: Few-shot lyrics transcription with self-training," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[25] L. Ou, X. Gu, and Y. Wang, "Transfer learning of wav2vec 2.0 for automatic lyric transcription," *arXiv preprint arXiv:2207.09747*, 2022.

[26] L. Yizhi, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, C. Xiao, C. Lin, A. Ragni, E. Benetos *et al.*, "Mert: Acoustic music understanding model with large-scale self-supervised training," in *The Twelfth International Conference on Learning Representations*, 2023.

[27] R. M. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. P. Bello, "Medleydb: A multitrack dataset for annotation-intensive mir research." in *ISMIR*, vol. 14, 2014, pp. 155–160.

[28] I. Loshchilov, F. Hutter *et al.*, "Fixing weight decay regularization in adam," *arXiv preprint arXiv:1711.05101*, vol. 5, 2017.

[29] S. Adavanne, A. Politis, and T. Virtanen, "Multichannel sound event detection using 3d convolutional neural networks for learning inter-channel features," in *2018 international joint conference on neural networks (IJCNN)*. IEEE, 2018, pp. 1–7.