FULL-EXCERPT INSTRUMENT RECOGNITION WITH FOUR LORA-ADAPTED AUDIO ENCODERS

Shuangrun Shao
Central Conservatory of Music, Beijing, China
25tw20@mail.ccom.edu.cn

ABSTRACT

We describe a clip-level multi-label instrument recognizer for MIREX 2026. Four pretrained encoders (MuQ, MERT-v2, PaSST and BEATs) process 10-second windows covering the entire excerpt. Per-encoder class queries attend to audio features; a learned weight for each encoder and instrument fuses their clip logits. Auxiliary frame activity and attention-based multiple-instance learning objectives complement mask-aware classification with partially observed labels. Low-rank adapters update the encoders while their base parameters stay frozen. Training combines OpenMIC-2018 with seven external instrument and audio datasets; unobserved tags in weakly annotated datasets do not become negative labels. The system emits the maximum window-level sigmoid confidence for every OpenMIC-based instrument class, scored at the fixed threshold 0.5. The training protocol combines strong and partially observed labels with dataset-balanced sampling and parameter-efficient encoder adaptation.

1. TASK AND ARCHITECTURE

MIREX 2026 evaluates instrument presence in music excerpts and ranks systems by hidden-set clip-level macro-F1 [1]. Each input maps to the 20-label OpenMIC-2018 vocabulary [2]; the organizers may exclude classes after a label audit. Scores are not post-hoc calibrated: given 10-second windows t, the decision for class c is $\hat{y}_c = \mathbf{1}[\max_t \sigma(z_{t,c}) \geq 0.5]$.

Audio is converted to mono and resampled to 32 kHz. Inputs up to 10 seconds use one window, with shorter recordings zero padded. Longer recordings use successive 10-second windows; the last window is aligned with the end and may overlap the preceding window. Four pretrained models provide complementary representations: the music-oriented MuQ [3] and MERT-v2-30s [4, 5], spectrogram-based PaSST [6], and general-audio BEATs [7]. MuQ and MERT-v2 internally receive 24 kHz waveforms, PaSST 32 kHz, and BEATs 16 kHz. For each branch, a learnable softmax-weighted sum of encoder hidden layers (when available) is projected to 512 dimensions. Twenty class-specific queries cross-attend to the branch's feature sequence, self-attend to model instrument co-occurrence, then produce window logits $z_{t,c}^{(b)}$. A learned softmax over branches is computed independently for each instrument:

$$z_{t,c} = \sum_{b=1}^{4} \frac{\exp a_{bc}}{\sum_j \exp a_{jc}} z_{t,c}^{(b)}. \quad (1)$$

Each branch also produces 20 frame-activity logits. A learned temporal attention pools those logits into weakly supervised clip predictions (MIL). The PaSST branch provides one clip embedding, treated as a length-one sequence; its activity head does not provide temporal localization. Only the fused query-head logits are used at inference. Class scores are pooled by a maximum over windows. Single-window outputs are unchanged; longer inputs require more inference and may cause additional false positives. There is no extra classifier, test-time ensemble or score calibration.

2. TRAINING DATA AND LOSS

The final training pool comprises OpenMIC-2018 [2], Slakh2100-redux [8], MusicNet acoustic recordings [9], GOOD-SOUNDS [10], RWC-Instruments [11], IRMAS [12], FSD50K [13] and MagnaTagATune [14]. OpenMIC's official train and validation subsets are included in final fitting; the official public test partition is excluded. Available train, validation and test subsets from the external datasets are included in the final fit. These external subsets and the OpenMIC validation subset are consequently *not* held-out tests for this trained system. The resulting manifest has 145,306 training excerpt records, of which 14,915 come from OpenMIC. All datasets are mapped into the same 20-class vocabulary; we retain the observation mask for each class and excerpt.

Slakh and MusicNet supply note-timed instrument activity. GOOD-SOUNDS and RWC-Instruments provide predominantly single-instrument recordings. These four datasets also supervise the frame-activity heads, while IRMAS provides only dominant-instrument positives, not frame-level ground truth. For FSD50K, the original training split retains recordings at most 10 seconds long; when the final fit includes longer clips from other FSD50K splits, a fixed 10-second crop is used. MagnaTagATune recordings use a fixed 10-second window. Clip tags do not guarantee an instrument occurs in that specific window.

© Shuangrun Shao. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** Shuangrun Shao, "Full-Excerpt Instrument Recognition with Four LoRA-Adapted Audio Encoders", submitted to *ISMIR LBD*, 2026.

For the weakly annotated datasets, only observed positive tags contribute to the loss: missing labels have mask zero. OpenMIC's partially annotated class labels use its official observation mask. The other strong datasets supply class negatives for their mapped vocabulary.

Training minimizes masked binary cross-entropy over the fused logits, plus per-branch frame-activity and attention-MIL terms:

$$\mathcal{L} = \mathcal{L}_{\text{clip}} + 0.5 \sum_{b} (\mathcal{L}^{(b)}_{\text{frame}} + \mathcal{L}^{(b)}_{\text{MIL}}). \quad (2)$$

Each observed class contributes only where its annotation mask is one; frame loss uses samples with frame targets. All four pretrained bases are frozen and rank-16 LoRA adapters [15] are inserted (scale 32, dropout 0.05). The run starts from pretrained encoders and randomly initialized prediction heads. It uses AdamW at learning rate 10^{-4}, a one-epoch warmup and cosine decay, 10 epochs, and 65,536 *with-replacement* draws per epoch. OpenMIC receives a fixed 10% quota, and the seven external datasets have equal relative sampling weight. A fixed draw count does not guarantee every distinct excerpt is visited.

3. REFERENCES

[1] MIREX, "2026: Audio Instrument Recognition," task specification, 2026. https://music-ir.org/mirex/wiki/2026:Audio_Instrument_Recognition.

[2] E. J. Humphrey, S. Durand, and B. McFee, "OpenMIC-2018: An open dataset for multiple instrument recognition," in *Proc. ISMIR*, 2018, pp. 438–444.

[3] H. Zhu *et al.*, "MuQ: Self-supervised music representation learning with Mel residual vector quantization," arXiv:2501.01108, 2025.

[4] R. Yuan *et al.*, "YuE2: Unifying symbolic and audio music generation at frontier quality," arXiv:2609.33757, 2026; "MERT-v2-30s" model card, https://huggingface.co/m-a-p/MERT-v2-30s.

[5] Y. Li *et al.*, "MERT: Acoustic music understanding model with large-scale self-supervised training," in *Proc. ICLR*, 2024.

[6] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient training of audio transformers with Patchout," in *Proc. Interspeech*, 2022.

[7] S. Chen *et al.*, "BEATs: Audio pre-training with acoustic tokenizers," arXiv:2212.09058, 2022.

[8] E. Manilow, G. Wichern, P. Seetharaman, and J. Le Roux, "Cutting music source separation some Slakh: A dataset to study the impact of training data quality and quantity," in *Proc. WASPAA*, 2019.

[9] J. Thickstun, Z. Harchaoui, and S. Kakade, "Learning features of music from scratch," in *Proc. ICLR*, 2017.

[10] O. Romani Picas and G. Bandiera, "A real-time system for measuring sound goodness," in *AES 138th Convention*, 2015.

[11] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC Music Database: Music Genre Database and Musical Instrument Sound Database," in *Proc. ISMIR*, 2003, pp. 229–230.

[12] 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 *Proc. ISMIR*, 2012, pp. 559–564.

[13] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "FSD50K: An open dataset of human-labeled sound events," *IEEE/ACM Trans. Audio Speech Lang. Process.*, 2022.

[14] E. Law, K. West, M. Mandel, M. Bay, and J. S. Downie, "Evaluation of algorithms using games: The case of music annotation," in *Proc. ISMIR*, 2009.

[15] E. J. Hu *et al.*, "LoRA: Low-rank adaptation of large language models," in *Proc. ICLR*, 2022.