Advanced Recombinant Transformer: Instrument-Level Audio Compatibility Scoring with Self-Supervised Acoustic Embeddings

Ahmad Hammoudeh*, Muhammad Taimoor Haseeb*, Gus Xia
Music X Lab, MBZUAI

*The first two authors contributed equally.

***Abstract*—Vertical loop compatibility—predicting the perceptual compatibility of two music loops layered together in a mix—has seen limited progress due to the absence of a large-scale, high-fidelity, labeled dataset. We introduce the Advanced Recombinant Transformer (ART). To train ART, we curate a large dataset of aligned positive pairs by rendering studio-quality stems from multi-instrument MIDI. We construct matched-condition negatives using score-based mining with calibrated thresholds based on a blinded musician audit. For representation, we replace log-mel features with frozen embeddings from the Music understanding model with large-scale self-supervised training (MERT), which capture timbral nuances and provide cues for higher-level musical structure. An order-invariant head scores pairwise compatibility. Experimental results show a 10.1% improvement in compatibility classification over the previous state-of-the-art. Extensive subjective tests (N = 107) on production-grade, human-composed audio loops find seamlessness and creativity ratings comparable to human-curated pairings. Samples are available at: https://loopcompatibility.github.io/demo.**

***Index Terms*—Neural Network Applications, Industrial Applications, Representation & Reasoning, Transformer Networks**

I. Introduction

Even though loops are standard in Digital Audio Workstations and have powered widely successful commercial tracks, automatically combining them in a musically coherent way remains non-trivial. Determining compatibility between pre-recorded loops requires capturing nuanced rhythmic, harmonic, and timbral interactions, making it a significant music information retrieval challenge [1]. This work focuses on *vertical* loop compatibility: deciding whether two loops can be layered together to compose a musically coherent blend. From a computational audio perception perspective, this requires predicting whether two signals can co-exist in a mixture without perceptual conflict.

While deep learning has shown considerable potential, progress is hampered by several challenges. First, the absence of large, high-quality datasets that include both compatible and incompatible loop pairs. Some previous approaches address this by decomposing tracks (non-negative tensor factorization (NTF) based loop extraction [2]; source-separated stem loops [3]), and treating within-track co-occurrences as positives. Standard source separation models typically produce four stems: vocals, bass, drums, and other. The *other* stem aggregates multiple instruments, producing composite rather than atomic loops, resulting in weak supervision for instrument loop compatibility learning. In addition, these outputs often suffer from noise, bleed, and distortion.

Negative sampling is especially challenging. To enforce loop incompatibility, prior methods often rely on artificial manipulations—loop reversal, beat shifting, or mismatched key/tempo. This risks training models to focus on superficial cues rather than true incompatibility, limiting generalization. Recent work uses AutoMashUpper to mine low-scoring pairs after matching their keys, tempo, and phase; improving realism, but offering no incompatibility guarantee [4], [5].

Feature representations used for modeling loop compatibility are another limitation. Earlier works rely on handcrafted audio features (e.g., chroma, Mel-frequency cepstral coefficients (MFCCs), etc.), while recent works employ magnitude mel-spectrograms of aggregated audio of loop pairs [4], [2], [3], [5]. Mel-spectrograms measure magnitude in the time–frequency domain rather than explicit musical structure. Models are forced to infer high-level semantics, like rhythmic patterns, harmonic progressions, and instrumentation, from energy patterns. Moreover, aggregating two loop audios into a single spectrogram obscures individual characteristics, making compatibility harder to model as a relational task. Motivated by the demonstrated transferability of self-supervised encoders such as wav2vec2 and HuBERT to downstream tasks with limited labeled data, we leverage frozen MERT embeddings to learn a compatibility function over loop pairs instead of aggregating signals into a single spectrogram [6], [7], [8].

We present Advanced Recombinant Transformer (ART), our compatibility model trained on a curated dataset produced by an audit-calibrated mining pipeline. We render studio-quality stems from multi-instrument MIDI to curate aligned positive pairs. Matched-condition hard negatives are mined by: (i) aligning key, tempo and phase of candidates; (ii) using a frozen Deep Recombinant Transformer (DRT) to obtain low-scoring pairs; and (iii) calibrating score cutoffs using a blinded musician audit [5]. We replace log-mel with frozen embeddings from MERT, a large self-supervised music encoder producing rich representations [8]. An order-invariant head scores pairwise compatibility. We evaluate on classification and retrieval metrics, test cross-domain generalization to human loops in zero-shot and fine-tuned settings, and conduct large-scale

listening tests (N=107). ART outperforms previous state-of-the-art by 10.1% higher accuracy, and listening tests rate outputs higher for musical seamlessness and creative fit.
Specifically, we make the following contributions:

- Studio-quality MIDI rendering to obtain instrument-level, aligned, positive loop pairs at scale. Hard negatives mined under matched key/tempo/phase and calibrated by professional audit to achieve ≥95% negative-label precision.
- Replacing log-mel inputs with MERT representations for loop compatibility modeling and evaluating multiple layer-selection strategies (including learned scalar mix).
- An order-invariant, transformer-based model for compatibility prediction (ART) demonstrating a 10.1% improvement over the previous state of the art (SOTA).
- We achieve state-of-the-art performance on compatibility classification and retrieval, cross-domain generalization to human recordings, and large-scale listening tests (N=107) showing human-comparable seamlessness and creativity.

The training and evaluation code and the list of loop pairs and associated metadata used in our experiments are publicly available at https://github.com/loopcompatibility/code. Due to licensing restrictions on certain commercial audio assets, we are unable to redistribute the raw-audio dataset. Instead, we provide scripts and documentation to reproduce the dataset where licensing permits.

II. Related Work

Davies et al. propose AutoMashUpper, segmenting tracks into downbeat-synchronous phrases, comparing chromagrams under key/tempo transforms, and rendering via time/pitch scaling [9]. They later improve efficiency and perceptual modeling with a faster chroma cross-correlation for harmonic similarity, and by incorporating rhythm and loudness into the estimation [4]. Lee et al. model vertical and horizontal compatibility with tempo, beat-synchronous chromagram, chord signatures, MFCCs, and volume. Candidate leads are evaluated against a Group of Background Units (GBU) using harmonic matching, harmonic change balance, and volume weighting [10]. Tsuzuki et al.'s Unisoner overlays vocal tracks from different singers of the same song on a common accompaniment by synchronizing each track using key-difference estimation and time-shift alignment [11]. Bernardes et al.'s Hierarchical Harmonic Mixing Method (HMM) extends Tonal Interval Space to audio to estimate dissonance and perceptual relatedness for small-scale compatibility, and key for large-scale compatibility to preserve diatonic structure over time [12]. MixMash builds a force-directed graph with onset density, spectral region, and MFCC timbre, enabling intuitive exploration of music collections [13].

Chen et al. present a learning-based loop-compatibility system: they extract loops from full mixes using an NTF-based loop-extraction algorithm as positives, and negatives from between-song sampling and within-song edits [2]. They compare a joint-input CNN with a Siamese model. The CNN excels at classification, whereas the Siamese model performs better on several ranking metrics [9]. Huang et al. model stem-group compatibility using MashupDB built via supervised separation [3]. Positives are original pairs, hard negatives use unmatched key/tempo with beat-phase offsets, and matched unlabeled pairs satisfy $|\Delta\text{key}| \leq 3$ with tempo in 0.8–1.2; two mel-CNNs are trained with label smoothing regularization for outliers (LSRO) semi-supervision. Haseeb et al. synthesize instrument-level loops from LMD to form positive pairs and mine negatives with matched attributes using AutoMashUpper [5]. While mining improves negative-sample realism, it does not guarantee incompatibility. Their audio is rendered with FluidSynth, yielding lower-fidelity stems than high-end libraries; fidelity significantly affects downstream learning and generalization [14]. Finally, they provide no ablations demonstrating generalization to human-composed loops.

III. Methodology

A. Data Pipeline

Modeling loop compatibility requires a dataset grounded in real production practice. However, existing datasets lack scale and rely on low-precision negatives created using superficial edits. We build a data pipeline to obtain over 73,000 attribute-aligned loop pairs, with high-quality labels (negative-label precision $\geq 95\,\%$), across 27 instrument roles and 11 genres, to enable strong vertical compatibility modeling.

Positive Loop Curation: Building on Haseeb et al., we generate positive samples by rendering Lakh MIDI Dataset (LMD) files into instrument-level tracks [5], [15]. From 175k+ MIDI files, we retain files with a single meter, tempo variance $< 5\%$, and length $\geq$ 32 bars, and segment each file into contiguous 8-bar windows w. For a track t in window w, we compute a track-coherence score $S_{\text{track}}(t, w)$ to decide whether the track behaves like a usable musical loop. Intuitively, the score rewards four properties: the onsets should lie close to the metrical grid, the rhythmic pattern should repeat from bar to bar, the track should be active for a meaningful portion of the window, and the onset density should fall in a musically plausible range. Formally, we compute $S_{\text{track}}(t, w)$ as:

$$\begin{aligned} S_{\text{track}}(t,w) = \tfrac{1}{4}\big(&S_{\text{grid}}(t,w) + S_{\text{bar}}(t,w) \\ &+ S_{\text{occ}}(t,w) + S_{\text{rate}}(t,w)\big) \end{aligned} \tag{1}$$

where $S_{\text{grid}}(t, w)$ is the fraction of onsets within a tempo-normalized tolerance $\tau = \max\{40\,\text{ms}, 0.07\,T_{\text{subdiv}}\}$ of the nearest n-subdivision gridline, with $n \in \{8, 12, 16\}$ and T_{subdiv} the per-beat subdivision duration; $S_{\text{bar}}(t, w)$ is the normalized autocorrelation at a one-bar lag of a bar-normalized onset-activation vector sampled at 16 subdivisions per beat; $S_{\text{occ}}(t, w)$ is the fraction of the window duration for which at least one note is active; and $S_{\text{rate}}(t, w)$ assigns full credit to onset rates in $[0.5, 12]\,\text{s}^{-1}$ and decreases linearly outside that range. A track is accepted when $S_{\text{track}}(t, w) \geq 0.60$. A window is accepted if it contains at least three accepted tracks and the median pitches of those tracks span at least two register bands (< 48, 48–72, and > 72 MIDI). A file is retained if it

contains at least three non-overlapping accepted windows. We then sample $N = 5{,}000$ files without replacement.

Manilow et al. show that higher-fidelity synthesis improves downstream learning and generalization [14]. We render the selected MIDI files with a deterministic offline renderer built in C++/JUCE, after MIDI normalization (pitch-bend consolidation, sustain-pedal folding, velocity bounding). MIDI tracks are assigned to instrument roles via confidence-scored heuristics (program, range, density), with low-confidence cases verified by humans. These tracks are then rendered into audio stems with commercial, role-specific sample libraries, primarily EastWest ComposerCloud, covering hundreds of instruments, using pinned presets. All stems pass through a minimal, fixed chain with true-peak control and are loudness-normalized to an EBU R128 target. Automated quality assurance (QA) removes silence, clipping, DC offset, and abnormal durations. Bar-aligned loop candidates are extracted, standardized to 10s (repeat/truncate), and near duplicates removed. We obtain a total of 38,253 valid positive loop pairs, providing 106 hours of positive sample data.

Negative Loop Mining: Most prior works construct Perturbation-Based Negatives via superficial edits, which may lead models to exploit low-level artifacts rather than learn true incompatibility. Haseeb et al. use reverse AMU scoring under matched attributes to mine presumable negatives, but their approach does not guarantee incompatibility [5], [9]. To mine negatives at scale, we use DRT as a frozen compatibility scorer. For each randomly drawn anchor loop, 200 candidates are sampled from the loop dataset, matched in key, tempo, and phase, and assigned compatibility scores. We retain up to ten lowest-scoring candidates per anchor as provisional negatives, subject to diversity constraints. Repeating this for 10,000 anchors and de-duplicating produces 69,794 provisional negative pairs. We conduct a blinded, stratified 20% audit in which professional musicians, unaware of pre-filtering, label each provisional negative pair as either compatible or incompatible. Using expert labels and DRT scores, we set per-stratum cutoffs τ_c as the largest thresholds for which $\Pr[\text{incompatible} \mid s \leq \tau_c] \geq 0.95$ (Wilson 95% CI lower bound ≥ 0.95). A symmetric exclusion band $\varepsilon = 0.02$ is applied: pairs with $s \leq \tau_c - \varepsilon$ are labeled *negative*, those with $|s - \tau_c| < \varepsilon$ discarded, and the rest *non-negative*. This yields a pool of matched-condition, musically incompatible negatives with high precision, reducing label leakage. After calibration, 34,897 negatives remain. The dataset is split into training, validation, and test sets; audited pairs, reserved as a gold-standard holdout, are not included in the training set.

TABLE I
SUMMARY OF DATASET COMPOSITION.

Split	Positive Pairs	Negative Pairs	Hours
Training	30,602	27,917	162.6
Validation	3,825	3,490	20.3
Test	3,826	3,490	20.3
Total	**38,253**	**34,897**	**203.2**

B. Self-Supervised Audio Representations

Self-supervised pretraining often matches or surpasses supervised learning across modalities: in NLP, masked-LM pretraining set then-SOTA results on GLUE and SQuAD [16], [17]; in speech, self-supervised encoders reach SOTA with limited labeled data [6], [7]. Motivated by this, we adopt MERT, a large self-supervised music encoder, as a frozen feature extractor [8]. Trained with a masked-prediction objective, MERT produces contextual, sequence-level embeddings indicative of pitch/tonal structure and rhythmic regularities. By contrast, mel spectrograms encode time–frequency information, rather than explicit musical structure, leaving models to infer higher-level attributes. Although we focus on MERT due to its music-oriented pretraining, ART is encoder-agnostic and can in principle use other Self-Supervised Learning (SSL) acoustic encoders as drop-in feature extractors. A systematic benchmark across different SSL encoders is left for future work. In our framework, the frozen MERT representations extracted for each loop are passed to ART, which models pairwise compatibility using order-invariant prediction heads.

C. Model Architecture

ART refers to the learned compatibility model trained on the curated loop-pair dataset produced by the data pipeline above. Loop pairs, already aligned in key, tempo, and phase, and loudness-normalized, are processed one at a time using MERT-330M to extract per-frame hidden states.

We obtain embeddings using four layer-selection strategies: (i) penultimate layer; (ii) last layer; (iii) uniform average of the last four; and (iv) a softmax-normalized scalar mix (one learned coefficient per layer). These choices span common ways of probing frozen self-supervised encoders. We compare the last and penultimate layers to test whether later or slightly earlier representations transfer better, use the last-four average as a simple fixed aggregation to smooth layer-specific noise, and include a scalar mix to learn a data-driven combination across layers. MERT is frozen in all cases. We train and evaluate two different classification heads: a multilayer perceptron (MLP) head and a Cross-Attention head across all four layer-selection strategies.

MLP head: We mean-pool the frame embeddings into a 1024-D vector. An order-invariant loop pair representation is formed by concatenating $(h_{\text{ref}}{+}h_{\text{cand}})$, $|h_{\text{ref}}{-}h_{\text{cand}}|$, $h_{\text{ref}}{\odot}h_{\text{cand}}$, and the cosine similarity of their ℓ_2-normalized vectors, resulting in $(3d{+}1) = 3073$-D input. A two-hidden-layer MLP with GELU activations and a dropout rate of 0.2 outputs a logit.

Cross-Attention head: Intuitively, bidirectional cross-attention lets each frame in one loop attend to frames in the other loop, and vice versa. This allows the model to compare the two sequences in both directions, so that each sequence representation is conditioned on the other before pooling and final compatibility prediction. Per-frame MERT sequences are down-sampled to $T{=}48$ and augmented with sinusoidal positions. Frames are projected $1024 \to d_m$ with $d_m{=}256$, processed by a single bidirectional multi-head Cross-Attention block (Pre-LN, 6 heads, dropout 0.1) followed by a residual

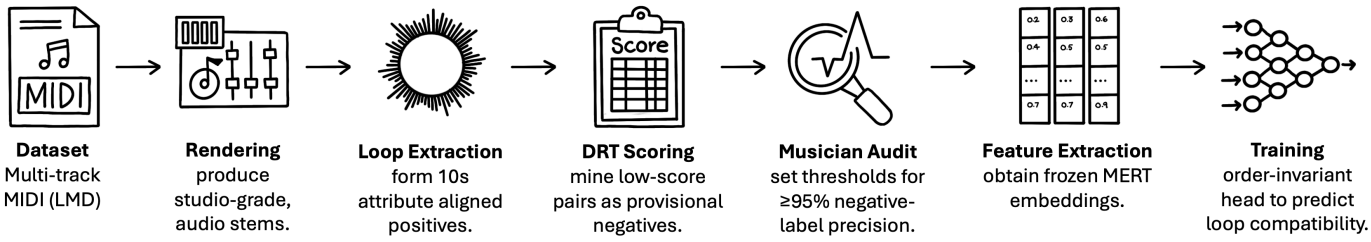

Fig. 1. High-level overview of our method — dataset curation pipeline and ART training for vertical audio loop compatibility.

feed-forward network (FFN), then mean-pooled to a 256-D loop vector with LayerNorm. The same order-invariant pair representation is constructed ($(3d_m{+}1) = 769$) and passed to a two-layer MLP classifier to produce the logit.

All variants are trained with binary cross-entropy (BCE) with logits loss and AdamW. We apply cosine learning-rate decay (10% warmup), gradient clipping at 1.0, early stopping, and set learning rate $\mathrm{lr}{=}10^{-4}$ for MLP head and $\mathrm{lr}{=}5 \times 10^{-4}$ for Cross-Attention.

IV. Evaluation

A. Objective Evaluation

We conduct ablations over four layer-selection strategies, each tested with pairwise heads. Each model is run with five random seeds; we report the mean results. At inference, logits are passed through a sigmoid to obtain $p(\text{comp})$. To disentangle gains from data quality versus MERT representations, we retrain DRT (log-mel inputs) using our curated dataset (DRT Base). Tables 2 and 3 summarize experimental results.

TABLE II
Classification results by model configuration.

Head	Accuracy (%) ↑	F1 Score ↑
MLP — Penultimate	87.6	88.2
MLP — Last	85.5	85.9
MLP — Last-4 avg	87.7	88.1
MLP — Scalar mix	88.7	89.3
Cross-Attention — Penultimate	91.4	92.1
Cross-Attention — Last	90.8	91.5
Cross-Attention — Last-4 avg	91.4	92.0
Cross-Attention — Scalar mix	**91.8**	**92.1**
DRT Base	81.6	83.5

Standard evaluation in the literature comprises classification accuracy and retrieval (ranking each query loop x's compatible partner x^+ among 100 attribute-matched candidates and reporting Avg. Rank). Results indicate Cross-Attention (scalar mix) performs best across both tasks; henceforth ART Base. Because DRT Base is retrained on the same curated dataset, the gap between ART Base and DRT Base highlights the contribution of frozen MERT embeddings and pairwise relational heads, beyond improvements due to dataset quality alone. Separately, while the negative labels are human audited, they are proposed using low-score mining, which can over-represent easy incompatibilities in the DRT scorer's tail. To address this, and to assess cross-domain generalization of ART Base to human recordings, we construct a second dataset using MoisesDB [18]. We curate 20k labeled pairs (10k pos, 10k neg), where negatives are sampled without conditioning on DRT score. ART Base is evaluated zero-shot on the MoisesDB test split, then fine-tuned on the MoisesDB training split and re-evaluated. Tables 4 and 5 show the method generalizes to human recordings, performs well zero-shot, and improves with fine-tuning (ART FT).

TABLE III
Retrieval results by model configuration.

Model	Avg. Rank ↓	Top 10 ↑	Top 30 ↑	Top 50 ↑
MLP — Penultimate	12.4	0.4	0.7	1.0
MLP — Last	14.6	0.4	0.7	1.0
MLP — Last-4 avg	13.4	0.4	0.7	1.0
MLP — Scalar mix	12.3	0.4	0.7	1.0
Cross-Attention — Penultimate	10.8	0.5	0.8	1.0
Cross-Attention — Last	11.9	0.4	0.8	1.0
Cross-Attention — Last-4 avg	10.2	0.5	0.8	1.0
Cross-Attention — Scalar mix	**9.7**	**0.5**	**0.8**	1.0
DRT Base	17.9	0.4	0.7	1.0

TABLE IV
Cross-domain classification results on MoisesDB.

Model	Accuracy (%) ↑	F1 Score ↑
ART Base	86.1	90.4
DRT Base	75.7	82.0
DRT (Original)	60.4	63.9
ART FT	**92.4**	**93.1**

B. Subjective Evaluation

We perform listening tests on Apple Loops[1] to assess usefulness in production. We evaluate three models: ART Base, ART FT, and DRT (Original). For comparison, we include a human-curated pairing and a random baseline. For each query loop, we construct a randomly sampled set of $N{=}99$ candidate loops from the same genre. Each model ranks the set after attribute alignment. We combine the query with

[1]https://support.apple.com/en-kz/guide/logicpro/lgcp734a05f6/mac

TABLE V
CROSS-DOMAIN RETRIEVAL EVALUATION ON MOISESDB.

Model	Avg. Rank ↓	Top 10 ↑	Top 30 ↑	Top 50 ↑
ART Base	18.5	0.4	0.6	0.9
DRT Base	24.6	0.3	0.4	0.8
DRT (Original)	43.0	0.2	0.2	0.5
ART FT	**11.3**	**0.4**	**0.7**	**1.0**

the model's top-ranked candidate to create a mix. We prepare $S{=}10$ evaluation sets, each containing five clips, and recruit $N{=}107$ listeners who report listening to music for at least 5 hours per week. For diversity, we sample across ages, genders, and backgrounds. Each participant rates five randomly drawn sets (25 clips, total 250 s of audio), on a 5-point Likert scale for Seamlessness (perceived continuity/cohesion) and Creativity (originality/inventiveness) of compositions. Figures 2 and 3 summarize our findings. The y-axis shows mean ratings; error bars denote the within-subject standard deviation derived from repeated-measures ANOVA [19]. ART outperforms the random baseline on both measures ($p < 0.05$) and other model baselines ($p < 0.001$) under repeated-measures analysis, and is comparable to the human-curated pairing.

V. CONCLUSION

We presented ART to address vertical loop compatibility, formulated as a pairwise audio matching and ranking problem. To enable training, we propose a data generation pipeline: filtering multi-instrument MIDI, rendering tracks into high quality audio stems, extracting loops, and mining matched-condition hard negatives via a frozen DRT calibrated by a blinded expert audit. We evaluate frozen MERT embeddings, as an alternative to log-mel inputs, scored by order-invariant heads. ART outperforms state-of-the-art on classification and retrieval tasks, and subjective tests indicate higher perceived seamlessness and creativity. ART generalizes to human recordings (MoisesDB); performs well zero-shot and improves further with modest fine-tuning. Our attribute-aligned negatives better reflect real-world production choices, and audit prunes borderline false negatives to increase label precision (negative-label precision $\geq$95%). Limitations remain. First, although MIDI-rendered audio is effective for large-scale pre-training, it cannot fully capture the timing, variability, and mix decisions present in human-produced loops. As a result, strong performance on the rendered dataset may still overestimate real-world performance on commercial or user-generated audio. Our additional MoisesDB and Apple Loops evaluations help assess this gap, but they do not eliminate it entirely. Second, automated rendering introduces curation risks, such as misclassified GM program numbers routing tracks to incorrect sample libraries; we mitigate this with confidence-scored role assignment and manual checks, but some labeling and rendering noise may persist. Third, MERT embeddings may reflect pretraining-corpus biases and are more compute-intensive than short-time Fourier transform (STFT)-based features, which may limit deployment in low-latency or resource-constrained settings. Finally, because negative labels are proposed using DRT-based mining, the resulting negative set can still be biased toward easier incompatibilities in the low-score tail of the DRT scorer, even after audit-based calibration. Future work includes broader evaluation across SSL acoustic encoders and contrasting genres and styles, score-independent negative set creation to reduce mining bias, and simultaneous modeling of vertical and horizontal loop compatibility for song-level arrangements. It also includes quantifying computational cost and deployment trade-offs, including latency, memory, and throughput, and engineering more compute-efficient methods for real-time applications.

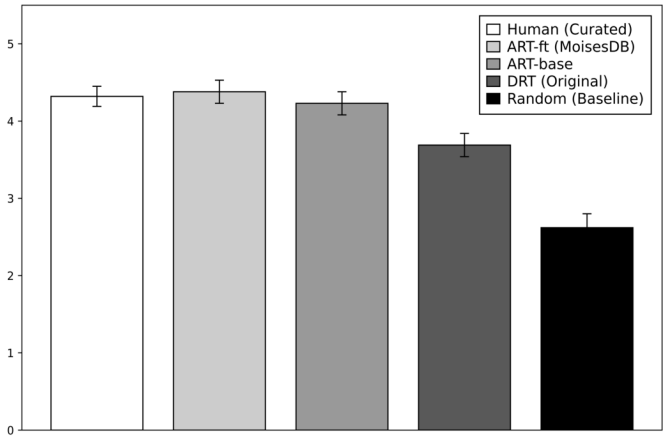

Fig. 2. Composition seamlessness (within-subjects ANOVA)

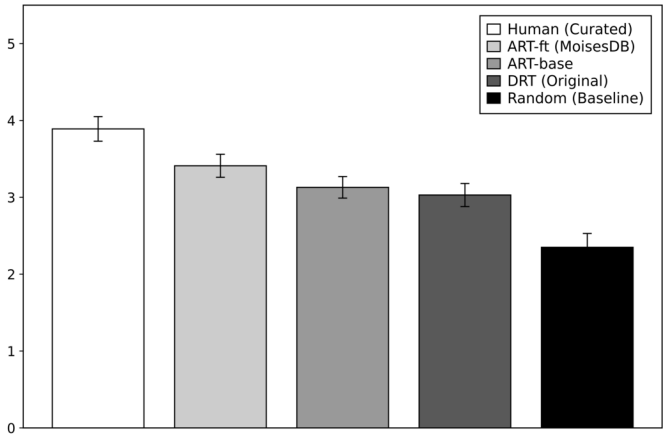

Fig. 3. Composition creativity (within-subjects ANOVA)

ACKNOWLEDGMENTS

The authors used ChatGPT and Grammarly for language editing (grammar, readability, phrasing) of portions of this manuscript. All suggested edits were reviewed and applied manually. [20], [21]

REFERENCES

[1] M. Goto, "Grand challenges in music information research," in *Multimodal Music Processing*, ser. Dagstuhl Follow-Ups, M. Müller, M. Goto, and M. Schedl, Eds. Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2012, vol. 3, pp. 217–226. [Online]. Available: https://drops-dev.dagstuhl.de/entities/document/10.4230/DFU.Vol3.11041.217

[2] B. Y. Chen, J. B. L. Smith, and Y.-H. Yang, "Neural loop combiner: Neural network models for assessing the compatibility of loops," *arXiv:2008.02011*, 2020.

[3] J. Huang, J. C. Wang, J. B. Smith, X. Song, and Y. Wang, "Modeling the compatibility of stem tracks to generate music mashups," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 1, May 2021, pp. 187–195.

[4] M. E. Davies, P. Hamel, K. Yoshii, and M. Goto, "Automashupper: Automatic creation of multi-song music mashups," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2014.

[5] M. T. Haseeb, A. Hammoudeh, and G. Xia, "Deep recombinant transformer: Enhancing loop compatibility in digital music production," in *Proceedings of the 25th International Society for Music Information Retrieval Conference, ISMIR 2024, San Francisco, California, USA, Nov 10-14, 2024*, 2024, pp. 890–896. [Online]. Available: https://doi.org/10.5281/zenodo.14877473

[6] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems*, 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html

[7] W. N. Hsu, B. Bolte, Y. 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. [Online]. Available: https://ieeexplore.ieee.org/document/9590303

[8] Y. Li, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, C. Xiao, C. Lin, A. Ragni, E. Benetos, N. Gyenge, R. B. Dannenberg, R. Liu, W. Chen, G. Xia, Y. Shi, W. Huang, Z. Wang, Y. Guo, and J. Fu, "Mert: Acoustic music understanding model with large-scale self-supervised training," in *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, Vienna, Austria, 2024. [Online]. Available: https://openreview.net/forum?id=w3YZ9MSlBu

[9] M. E. P. Davies, P. Hamel, K. Yoshii, and M. Goto, "Automashupper: An automatic multi-song mashup system," in *International Society for Music Information Retrieval Conference*, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:28117

[10] C. L. Lee, Y. T. Lin, Z. R. Yao, F. Y. Lee, and J. L. Wu, "Automatic mashup creation by considering both vertical and horizontal mashabilities," in *International Society for Music Information Retrieval Conference*, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:17802326

[11] K. Tsuzuki, T. Nakano, M. Goto, T. Yamada, and S. Makino, "Unisoner: An interactive interface for derivative chorus creation from various singing voices on the web," in *ICMC*, 2014.

[12] G. Bernardes, M. Davies, and C. Guedes, "A hierarchical harmonic mixing method," in *Music Technology with Swing*, ser. Lecture Notes in Computer Science, vol. 11265. Cham: Springer, Cham, 2018, cMMR 2017. [Online]. Available: https://doi.org/10.1007/978-3-030-01692-0_11

[13] C. Maçãs, A. Rodrigues, G. Bernardes, and P. Machado, "Mixmash: A visualisation system for musical mashup creation," in *2018 22nd International Conference Information Visualisation (IV)*, 2018, pp. 471–477.

[14] E. Manilow, G. Wichern, P. Seetharaman, and J. L. Roux, "Cutting music source separation some slakh: A dataset to study the impact of training data quality and quantity," in *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE, 2019, pp. 45–49.

[15] C. Raffel, "Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching," Ph.D. dissertation, PhD Thesis, 2016.

[16] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*. Minneapolis, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423/

[17] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," 2019. [Online]. Available: https://arxiv.org/abs/1907.11692

[18] I. Pereira, F. Araújo, F. Korzeniowski, and R. Vogl, "Moisesdb: A dataset for source separation beyond 4-stems," 2023.

[19] H. Scheffe, *The Analysis of Variance*. John Wiley & Sons, 1999, vol. 72.

[20] OpenAI, "Chatgpt," https://chat.openai.com, 2024, large language model.

[21] Grammarly Inc., "Grammarly," https://www.grammarly.com, 2024, aI-powered writing assistant.