

THE RHYTHM IN ANYTHING: AUDIO-PROMPTED DRUMS GENERATION WITH MASKED LANGUAGE MODELING

Patrick O'Reilly¹ Julia Barnett¹ Hugo Flores Garcia¹
 Annie Chu¹ Nathan Pruyne¹ Prem Seetharaman² Bryan Pardo¹
¹Northwestern University, Evanston, USA ²Adobe Research, San Francisco, USA

patrick.oreilly2024@u.northwestern.edu

ABSTRACT

Musicians and nonmusicians alike use rhythmic sound gestures, such as tapping and beatboxing, to express drum patterns. While these gestures effectively communicate musical ideas, realizing these ideas as fully-produced drum recordings can be time-consuming, potentially disrupting many creative workflows. To bridge this gap, we present TRIA (The Rhythm In Anything), a masked transformer model for mapping rhythmic sound gestures to high-fidelity drum recordings. Given an audio prompt of the desired rhythmic pattern and a second prompt to represent drumkit timbre, TRIA produces audio of a drumkit playing the desired rhythm (with appropriate elaborations) in the desired timbre. Subjective and objective evaluations show that a TRIA model trained on less than 10 hours of publicly-available drum data can generate high-quality, faithful realizations of sound gestures across a wide range of timbres in a zero-shot manner.

1. INTRODUCTION

Sound gestures such as tapping and beatboxing provide a convenient and idiomatic means of expressing rhythmic ideas. Rather than “literally” specifying a rhythmic idea through one-to-one imitation, sound gestures often capture a reduced, high-level representation of the desired rhythm—for instance, a beatboxer may only voice one element where many have simultaneous onsets, or leave certain elements unvoiced and implied. Realizing these gestures as fully-produced drum arrangements often requires many steps: the voiced sound elements in a gesture must be mapped to appropriate drum parts, unvoiced or implied elements must be plausibly reconstructed, the resulting arrangement must be performed and recorded or sequenced and synthesized digitally in audio editing software, and the final recording may require further processing to shape the timbre satisfactorily. By contrast, many creative workflows

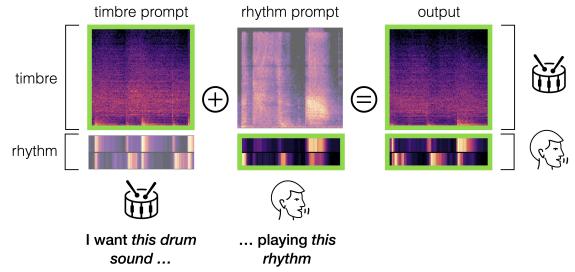


Figure 1. TRIA conditions generation of a new drum recording on two prompts: the timbre of an example drum recording (illustrated by a spectrogram), and the rhythm of a sound gesture (the dualized features in Section 3.1).

may benefit from the ability to rapidly generate diverse full-drumkit realizations of rhythmic sound gestures.

To bridge this gap, we propose TRIA (The Rhythm In Anything), a masked transformer model for mapping arbitrary rhythmic sound gestures to high-fidelity drum recordings. Given two audio prompts—one specifying the basic desired rhythm via a sound gesture, and one specifying the desired drum timbre via an example recording—TRIA synthesizes full-drumkit audio playing a fleshed-out arrangement of the desired rhythm in the desired timbre. TRIA can faithfully realize sound gestures in unseen timbres in a zero-shot manner despite its relatively small model size (43M trainable parameters) and training dataset (less than 10 hours of publicly-available drum recordings from MusDB18-HQ [1]). Through both quantitative comparisons and qualitative human listening evaluations, we demonstrate that TRIA matches or exceeds the performance of a 1-billion parameter state-of-the-art model [2] trained on 20,000 hours of public and private data in converting sound gestures to drum recordings.

Our contributions are as follows:

1. A model capable of mapping arbitrary rhythmic sound gestures to high-fidelity drum recordings using drum timbres specified at inference time
2. A dualized representation that lets the model capture salient rhythmic structure across drum and non-drum sound classes
3. Subjective and objective evaluations showing the importance of the dualized representation and the

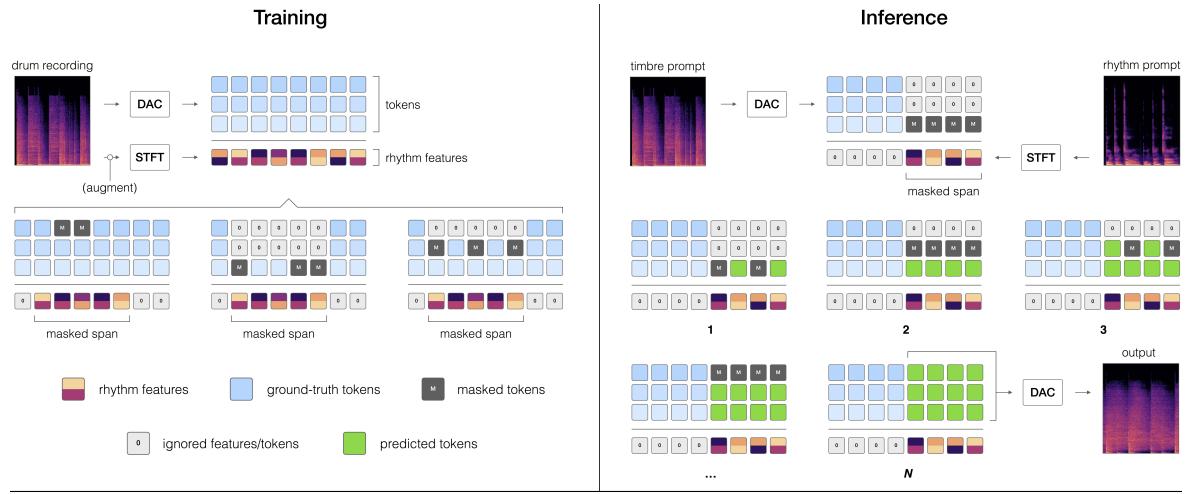


Figure 2. The proposed TRIA system. During training (left), acoustic tokens of a tokenized drum recording are predicted, conditioned on surrounding unmasked tokens and rhythm features extracted from an augmented version of the recording; we illustrate three training examples. During inference (right), we fix the timbre prompt as a prefix and predict a masked suffix conditioned on aligned features extracted from the rhythm prompt. Inference predicts tokens in coarse-to-fine order.

model’s ability to generate musically-pleasing translations that adhere to rhythm and timbre prompts

We provide audio examples and code on our webpage.¹

2. RELATED WORK

The translation of simple rhythmic gestures into full drum beats has been explored in the symbolic domain, notably in the GrooVAE models proposed by Gillick et al. [3]. While these allow for mapping single-voice MIDI drum patterns to full-drumkit expressive MIDI performances, they do not allow for audio-prompted rhythm or timbre specification.

In the audio domain, Santos & Cardoso applied RAVE [4] models to a tap-to-drums translation task [5]. However, RAVE does not support zero-shot audio-prompted timbre specification, but instead requires re-training for each new specified timbre. In general, recent neural network-based timbre transfer systems are similarly constrained or else support only pitched instruments [6, 7]. One exception is MelodyFlow [2], which performs text-guided audio editing via latent diffusion inversion, hypothetically allowing for the specification of arbitrary timbres via text prompts. We perform extensive comparisons between our proposed system and MelodyFlow in Section 4.

A number of systems use transcription to translate beatbox audio into drum recordings via synthesis from a predicted MIDI representation, but generally require user-specific calibration for accurate transcription and do not support audio-prompted timbre specification [8, 9]. In general, transcription-based systems are constrained to narrow sound gesture types with well-defined audio-symbol mappings or available annotated data for supervised training (e.g., beatboxing), and limited to “literal” mappings of timbres onto atomic sound events. By contrast, we propose an audio-prompted, *self-supervised* approach for

mapping simple rhythmic gestures to potentially complex full-drumkit recordings, allowing for the generation of arrangement details not explicit in the rhythm gesture.

Previous works have hypothesized that musicians often perceive and arrange drum patterns using implicit two-voice “dualized” representations that oscillate between low and high states [10, 11]. However, the use of dualized representations for music generation has been limited to the symbolic domain [12]. Our proposed system obtains dualized representations from audio (Section 3.1) to guide the generation of drum audio, letting us specify rhythmic structure with non-drum sounds (e.g. finger tapping).

Finally, our work differs from prior work on generating symbolic rhythm patterns [13–15], drum loops [16, 17], and drum samples [18–20] in that we seek to convert sound gestures into audio-domain full drumkit performances.

3. METHOD

We next describe the design of the proposed TRIA system.

Architecture: Similar to VampNet [21], TRIA is a transformer-based masked language model. TRIA consists of 12 standard transformer blocks, each with hidden size $h = 512$, 8 attention heads, and rotary positional encoding [22], resulting in 43 million trainable parameters.

Audio Tokenization: TRIA predicts acoustic tokens produced by Descript Audio Codec (DAC) [23]. Within DAC, audio is segmented into a series of T frames, each of which is mapped to a vector representation via a fully convolutional encoder. Encoded vectors are quantized with a hierarchical sequence of C vector-quantizers, each with its own codebook. Each quantizer encodes the residual between the original and the quantized representation produced by the previous quantizers. Quantized vectors are represented by their codebook indices, resulting in a token representation of C codebooks by T frames. A matched decoder converts $C \times T$ token representations into audio.

¹ <https://therhythminanything.github.io/>

Masked Language Modeling: TRIA generates drum audio by predicting missing or “masked” DAC tokens within a partially-masked “buffer” of size $C \times T$, conditioned on unmasked tokens (representing the target timbre and generated content), as is typical for masked token modeling. TRIA, however, also conditions generation on aligned rhythm features representing the target rhythm (see Section 3.1). Once all masked tokens are predicted, they are mapped to 44.1kHz mono audio via the DAC decoder.

To produce predictions for masked tokens in the buffer within a specific codebook $c \in [0, C - 1]$, all tokens in the buffer are first mapped to continuous vectors of size h via separate learned embedding tables per codebook, with masked tokens mapped to a single learned “mask” embedding shared across all codebooks. Recall that the tokens in every codebook at level $c' > c$ correct the residual error of the token at level c . Therefore, if a token at level c is masked, all corresponding embedding vectors in codebooks $c' > c$ are zeroed. Embedding vectors are then summed across codebooks to obtain a sequence of shape $h \times T$. Rhythm features (Section 3.1) are projected to the hidden dimension and zeroed for frames in which there are no masked tokens, resulting in a corresponding conditioning sequence of shape $h \times T$. The two sequences are summed and passed to the transformer, which predicts a probability distribution over tokens in codebook c at each frame via one of C codebook-specific projection layers.

Inference: At inference, we take as inputs a timbre prompt (drum) recording and a rhythm prompt (sound gesture) recording. We construct a buffer in which the tokenized timbre prompt serves as an unmasked prefix, with all subsequent frames (corresponding to the length of the rhythm prompt) fully masked. We compute rhythm features aligned to this masked suffix from the rhythm prompt.

We then perform SoundStorm-style inference [24] to iteratively predict masked tokens in each codebook in coarse-to-fine order, using the schedule of Chang et al. [25] to gradually unmask or “confirm” tokens in the suffix. We adopt temperature-based nondeterministic unmasking from VampNet and causal bias from StemGen [26] to favor unmasking earlier tokens in the buffer first.

Thus, we fill in the masked suffix using timbral information from the timbre prompt and rhythmic information from the rhythm prompt, resulting in a generation that adheres to both prompts. By specifying the number of inference iterations over which each codebook is unmasked, we can expend more compute on challenging high-entropy early generation steps and less on highly-determined later steps. For all experiments reported in this paper, we use an inference schedule of $\{8, 8, 8, 8, 8, 4, 4, 4, 4\}$ iterations for DAC’s 9 respective codebooks in coarse-to-fine order, classifier-free guidance [27] weight 2.0, unmasking temperature 10.0, and causal bias 1.0.

Training: At each training iteration, we sample a drum recording that serves as both timbre and rhythm prompt, tokenizing with DAC to obtain a buffer and computing rhythm features at a matching temporal resolution. We select a random codebook and a random span of consec-



Figure 3. Results of the listener preference evaluation detailed in Section 4.3. We plot win rates for TRIA and MelodyFlow generations from rhythm prompts sampled from the AVP and TapTamDrum (TTD) datasets, as well as random anchors from MoisesDB drums.

utive frames covering 50% to 75% of the buffer length, and mask a subset of tokens within this codebook and span according to the cosine schedule proposed by Chang et al. [25]; we then compute cross-entropy loss between TRIA’s predicted distributions at masked token positions and the corresponding ground-truth tokens. To allow TRIA to process rhythm prompts from a variety of sound sources and recording conditions, we apply noise, band-pass filtering, pitch shift, phase shift, and equalization to the rhythm prompt audio with independent 25% probabilities at each iteration. To provide control over the degree of adherence to the rhythm prompt, we implement classifier-free guidance [27] by zeroing rhythm features in 20% of training iterations to learn unconditional mappings, and then performing weighted interpolation between unconditional and conditional predictions at inference time.

We train all TRIA models on drums from a 90% split of the MusDBHQ-18 dataset [1], totaling 8 hours of audio. We train on 6-second random excerpts for 100,000 iterations at a batch size of 48 on 4× NVIDIA A10G GPUs, requiring ~ 27 hours per model. Training and inference are illustrated in Figure 2.

3.1 Dualized Rhythm Representation

To allow inference on arbitrary sound gestures while training only on drum audio, we require (1) timbre-rhythm disentanglement, with timbre information for the prediction of masked token spans provided by unmasked tokens outside the span and rhythm information provided by aligned rhythm features within the span; and (2) a rhythm feature representation that captures the structure of both drums and sound gestures with vastly different frequency energy distributions. If timbre-rhythm disentanglement is not enforced, e.g. if timbre information leaks from the rhythm features, TRIA will not apply the specified timbre. If there exists a modality gap between drums and sound gestures within the rhythm feature representation, TRIA will struggle to map sound gestures to plausible drum generations.

The simplest rhythm representation satisfying these criteria is a one-dimensional sequence of loudness estimates, which captures onset information similar to GrooVAE [3].

Model	F1 Snare ↑		F1 Kick ↑	
	30ms	100ms	30ms	100ms
Random anchor	0.04	0.15	0.09	0.29
MelodyFlow _{0.0}	0.08	0.16	0.11	0.19
MelodyFlow _{0.1}	0.11	0.13	0.13	0.18
MelodyFlow _{0.2}	0.19	0.23	0.21	0.23
TRIA _{1Band}	0.23	0.35	0.38	0.50
TRIA _{2Band} *	0.32	0.47	0.52	0.66
TRIA _{2Band-NA}	0.10	0.17	0.47	0.62
TRIA _{3Band}	0.33	0.50	0.61	0.71
TRIA _{4Band}	0.30	0.47	0.59	0.72

Table 1. F1 scores of automatic snare and kick transcriptions of MelodyFlow and TRIA generations from annotated AVP beatbox recordings at 30ms and 100ms onset tolerances. Higher scores indicate generations preserve the placement of kicks and snares from beatbox recordings.

However, researchers have found that onset representations fail to adequately capture relationships between multiple elements within percussion patterns and human sound gestures [10, 11] – for instance, distinct kick and snare vocalizations within a beatbox recording may be “flattened” into indistinguishable loudness spikes, making it difficult for TRIA to faithfully map the beatbox to drums. On the other hand, if the rhythm feature representation is too fine-grained, e.g. a full spectrogram, it will leak timbre information from the rhythm prompt and cause TRIA to ignore the timbre prompt. Additionally, drums and sound gestures will likely manifest distinctly in fine-grained feature representations, causing a train-inference mismatch.

To address these potential pitfalls, we propose a rhythm feature representation based on a two-band spectrogram with an adaptive splitting frequency. We start with an 80-bin mel-spectrogram of the rhythm prompt audio and compute a splitting frequency that equally divides energy into low and high bands, summing all bins within each band. We then standardize each band independently, apply a sigmoid nonlinearity to bound all values to $[0, 1]$, and quantize all values to 33 steps ($0, \frac{1}{32}, \frac{2}{32}, \dots, 1$) within this range.

Our motivation for this representation is twofold. First, a two-voice representation allows core elements of drum recordings and sound gestures to manifest distinctly, but lacks sufficient detail to leak timbre information or distinguish between drum recordings and sound gestures. Second, two-voice or “dualized” rhythm representations have been explored previously for the analysis and generation of drum patterns in the symbolic domain [10–12]. We extend this line of inquiry by evaluating the efficacy of audio-derived dualizations for audio generation.

4. EXPERIMENTS

We empirically validate TRIA’s ability to map sound gestures to full-drumkit recordings in user-specified timbres across two specific sound gesture types (beatboxing and

Model	MFCC-Sim		
	Rhythm ↓	Timbre ↑	Random
MelodyFlow _{0.0}	0.88	--	0.81
MelodyFlow _{0.1}	0.92	--	0.86
MelodyFlow _{0.2}	0.96	--	0.85
TRIA _{1Band}	0.85	0.95	0.87
TRIA _{2Band} *	0.85	0.96	0.87
TRIA _{2Band-NA}	0.83	0.93	0.85
TRIA _{3Band}	0.86	0.95	0.87
TRIA _{4Band}	0.84	0.96	0.86

Table 2. Timbral similarity between model outputs, input rhythm/timbre prompts, and random drum recordings as measured by time-averaged MFCC cosine similarity. Higher-than-random similarity with the rhythm prompt implies timbre leakage, while higher-than-random similarity with the timbre prompt implies prompt adherence.

tapping). We conduct both subjective human evaluations and objective evaluations of generation quality and adherence to rhythm and timbre prompts.

4.1 Models

TRIA: In addition to the TRIA system described in Section 3 (TRIA_{2Band}*), we validate our choice of rhythm feature representation by comparing variants of TRIA trained on 1-band (TRIA_{1Band}), 2-band with no adaptive frequency split (TRIA_{2Band-NA}), 3-band (TRIA_{3Band}), and 4-band rhythm features (TRIA_{4Band}).

MelodyFlow: we compare TRIA to MelodyFlow [2], a state-of-the-art text-prompted music editing system. MelodyFlow can apply text-specified timbres to sound gestures using regularized latent inversion, which maps an encoded sound gesture to an initial noise estimate and then resynthesizes it conditioned on the text prompt via flow-matching. This is done by a 1-billion parameter transformer model trained on a mix of private and licensed music totalling 20,000 hours. The degree to which MelodyFlow preserves the structure of the rhythm prompt can be coarsely controlled by specifying the “target flow step” for inversion, with 0.0 corresponding to full noising and 1.0 corresponding to no noising (where the audio is left unaltered). In our experiments we compare target flow steps of 0.0, 0.1, and 0.2 (MelodyFlow_{0.0}, MelodyFlow_{0.1}, and MelodyFlow_{0.2}, respectively); we find that higher values result in negligible adherence to the specified timbre. We use the default settings of 128 inference steps, “Euler” solver, and ReNoise [28] regularization strength 0.2. To allow fair comparisons with TRIA, we downmix MelodyFlow generations from stereo to mono and downsample from 48kHz to 44.1kHz.

4.2 Datasets

We evaluate both TRIA and MelodyFlow on rhythm prompts drawn from two datasets of sound gestures: AVP

Model	KAD _{PANN} ↓	KAD _{CLAP} ↓
TRIA _{1Band}	6.95	6.81
TRIA _{2Band} *	4.56	5.05
TRIA _{2Band-NA}	6.61	10.63
TRIA _{3Band}	4.53	5.46
TRIA _{4Band}	4.14	4.61

Table 3. Kernel Audio Distance (KAD) between a set of 500 generations from each model and a reference distribution of 500 drum excerpts from MoisesDB; lower scores indicate better audio quality.

[29], containing 56 amateur beatbox improvisations across 28 participants and 2 conditions with human-annotated transcriptions; and TapTamDrum [11], containing 1116 two-tone tapping imitations of drum beats across 4 participants. To avoid overlap with TRIA’s training data, we sample audio timbre prompts from the MoisesDB dataset [30], which contains drum stems from 240 commercial-quality music tracks. Because MelodyFlow requires timbre specification via text rather than audio prompts, we generate 50 descriptions of acoustic and electronic drum kit timbres using GPT-4.5 [31] which we manually inspect to ensure quality and diversity. Due to the lack of available drumkit timbre description datasets and our difficulty in obtaining diverse captions from drum audio using existing multimodal models [32], we settle on these synthetic descriptions as a reasonable approximation of “plausible” text prompts, and consult with the MelodyFlow authors to ensure descriptions are formatted appropriately for the model. In all experiments, we sample 2-second timbre prompts for TRIA and generate from rhythm prompts trimmed to a maximum duration of 4 seconds.

4.3 Subjective Evaluation

We first aim to understand how human listeners rate TRIA’s translations of sound gestures to drums when compared to the state-of-the-art model MelodyFlow. To this end, we conduct a listening evaluation utilizing RESEval [33], a framework for subjective evaluation tasks on crowdworker platforms; we recruit evaluators through the online research platform Prolific². We evaluate the TRIA_{2Band}* and MelodyFlow_{0.2} variants, as we find that these models provide a good balance of adherence to both rhythm and timbre prompts.

Data Preparation: We prepared 80 sets of short (3–4 second) audio clips. Each set contained (1) a reference sound gesture serving as a rhythm prompt, drawn either from the AVP “personal” condition (beatboxing) or TapTamDrum (tapping); (2) a TRIA generation from the rhythm prompt; (3) a MelodyFlow generation from the rhythm prompt; and (4) a random MoisesDB drum excerpt, unrelated to the rhythm prompt, as a low anchor. We generated these 80 sets using 10 rhythm prompts (5 beatboxing, 5 tapping) and 8 timbre prompts per rhythm prompt.

TRIA’s audio timbre prompts were drawn randomly from MoisesDB drum excerpts, while MelodyFlow’s text timbre prompts were drawn from the aforementioned set of 50 generated timbre descriptions. To ensure broadly comparable timbres across generations, we restricted our audio timbre prompts to acoustic drum kit recordings and our text prompts to descriptions of acoustic drum kit timbres.

ABX Trials: We leveraged the findings of Cartwright et al. [34, 35] and deployed pairwise comparison evaluations using remote crowdworkers. In our study, crowdworkers performed ABX trials: they heard a reference rhythm prompt (“X”) and were randomly presented with two clips (“A” and “B”) from the corresponding (1) TRIA generation, (2) MelodyFlow generation, or (3) a random drum excerpt to act as a low anchor. They were then asked to select “A” or “B” given the criteria:

Select which of the two choices is a more musically pleasing translation from the reference clip to drums that captures the original rhythm and groove of the reference clip.

Full coverage of our 80 sets required 3 pairwise comparisons per set: TRIA vs. MelodyFlow, TRIA vs. Random Excerpt, and MelodyFlow vs. Random Excerpt. We required 5 listeners evaluate each comparison, resulting in $80 \times 3 \times 5 = 1200$ total trials. From our ABX results, we computed the win rate of TRIA and MelodyFlow on each dataset and evaluated the statistical significance of the indicated listener preference. We present the results of our subjective evaluation in Figure 3.

Participant Recruitment: We recruited 120 US English-speaking human listeners with an approval rating of $\geq 95\%$ and a record of completing 100+ prior tasks on Prolific. Each listener evaluated 10 randomly assigned ABX pairwise comparisons. To ensure data quality, participants had to pass a listening test assessing tone sensitivity from 55 Hz - 10 kHz [36], along with attention checks. They were paid \$2.50 per set of 10 comparisons, estimated to be equivalent to \$18.75/hour. We excluded participants who failed the listening test, as well as those who preferred the Random Excerpt $\geq 80\%$ of the time, as this suggests they disregarded the given evaluation criterion of rhythm adherence. Following data cleaning, we had 116 participants with a total of 1160 evaluation pairs.

4.4 Rhythm Prompt Adherence

To evaluate TRIA’s preservation of the rhythmic structure of sound gestures when translating to drums, we conduct an automated transcription evaluation. We sample 250 generations from each MelodyFlow and TRIA variant conditioned on rhythm prompts drawn from the AVP beatbox dataset, all of which have ground-truth human annotations of kick drum, snare, and hi-hat vocalizations. We then transcribe these generations using the pretrained “Frame-RNN” drum transcription model of Zehren et al. [37]. Finally, we measure the correspondence between transcribed and ground-truth kick and snare drum parts using the onset F1 score with 30ms and 100ms tolerances, as is common in

² <https://www.prolific.com/>

the drum transcription literature [38, 39]. Higher F1 scores indicate tighter correspondence between the kick and snare vocalizations in the rhythm prompt and the kick and snare drums in the generated audio. We report results in Table 1.

4.5 Timbre Prompt Adherence

To evaluate TRIA’s treatment of timbre information, we compute the cosine similarity between time-averaged 80-dimensional MFCC representations of the generated audio and timbre prompt (indicating adherence to the timbre prompt), the generated audio and rhythm prompt (indicating the degree of timbre leakage from the rhythm prompt), and the generated audio and a random excerpt from MoisesDB drums (as an anchor). While this measurement of spectral correspondence provides a coarse approximation of timbral similarity, we find that it captures strong trends in each model’s treatment of timbre. Because MelodyFlow allows timbre specification through a text prompt, not an audio prompt, we can only compute its output similarity to the rhythm prompt and random excerpt. Our results are reported in Table 2. We further illustrate the processing of rhythm and timbre prompts by both systems in Figure 4.

4.6 Audio Quality

To evaluate the realism of generated audio, we compute the Kernel Audio Distance (KAD) [40, 41] between 500 outputs from each method and a reference distribution of 500 random excerpts of MoisesDB drums. Similar to Fréchet Audio Distance (FAD) [42], KAD measures the similarity of the generated distribution to a reference distribution, while showing stronger alignment with human quality ratings less bias at small sample sizes. For KAD we consider the “PANN” embedding variant [43], which the authors show is most correlated with human perception, and the “CLAP-Laion-Music” embedding variant [44], which leverages a model trained specifically on music. Because TRIA receives audio timbre prompts from the reference distribution while MelodyFlow receives text timbre prompts, we compare only variants of TRIA for fairness. We report results in Table 3.

5. DISCUSSION

Our experimental results demonstrate TRIA’s efficacy in translating rhythm gestures to full-drumkit recordings faithful to the rhythm and timbre prompts. As illustrated in Figure 3, our subjective evaluation shows no statistically significant preference between TRIA and MelodyFlow generations. *This is promising given that MelodyFlow is roughly 25× the size, and trained on 2,000× the data.* Additionally, both models are strongly preferred to random drum excerpts at significance $p \leq 0.001$ according to a two-sided binomial test, indicating that both succeed in capturing the core groove and structure of rhythm prompts.

The results of our transcription evaluation, presented in Table 1, show that TRIA strongly outperforms MelodyFlow in preserving the rhythmic structure of beatbox sound gestures as indicated by correspondence of kick

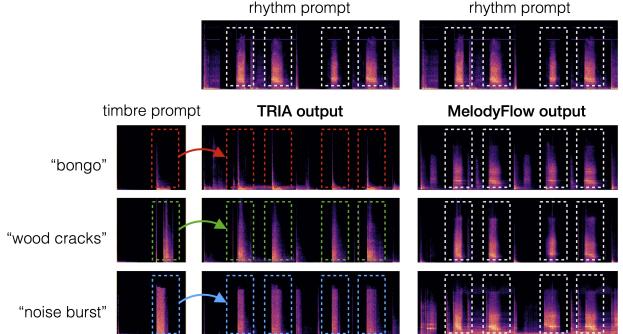


Figure 4. Given a rhythm prompt (top) with vocal kick and snare drum imitations, the “snare” sound can be replaced by user-provided samples via TRIA’s timbre prompting ability: (a) a bongo drum, (b) wood cracks, and (c) a noise burst. Given corresponding timbre prompts in text form, MelodyFlow adheres more closely to the spectral content of the rhythm prompt.

and snare placement in the rhythm prompt and generated audio. While increasing the target flow step improves MelodyFlow’s rhythm adherence slightly, it still significantly underperforms all evaluated TRIA variants. *These results demonstrate the strength of TRIA’s dualized rhythm feature representation, which outperforms both a 1-band representation and a non-adaptive 2-band representation that naively splits the mel spectrogram along its center frequency.* Adaptive 3- and 4-band rhythm feature representations yield diminishing returns as they slightly increase the accuracy of kick placement, but do not have a meaningful effect on snare placement. This indicates that a dualized representation may be sufficient to capture the core rhythmic structure of many sound gestures, while single-voice representations are likely insufficient.

The results of our timbre evaluation, presented in Table 2, show that TRIA generations exhibit lower spectral correlation with the rhythm prompt than random anchors, and higher correlation with the timbre prompt than random anchors – *indicating both a lack of timbre leakage from the rhythm prompt and strong adherence to the timbre prompt.* In contrast, MelodyFlow generations exhibit higher-than-random spectral correlation with the rhythm prompt, indicating timbre leakage. We provide examples illustrating these behaviors in Figure 4: MelodyFlow often mimics the spectral structure of rhythm prompts, while TRIA effectively utilizes a diverse array of timbre prompts to determine spectral structure. This audio-prompted timbre mapping is a key advantage of TRIA over text-prompted systems, allowing for more specific exemplar-based steering of generations. Finally, as shown in Table 3, *our dualized rhythm features outperform both 1-band and non-adaptive 2-band features in producing realistic drum audio.*

Overall, these results show the promise of our proposed approach even in small model and data regimes. Directions for future work include scaling the model and dataset; leveraging TRIA’s existing capabilities for other inference paradigms such as inpainting and drums-to-drums conversion; and exploring learnable dualized rhythm features.

6. ACKNOWLEDGEMENTS

This work was supported by NSF Award Number 2222369. We would also like to thank André Carvalho dos Santos and Gaël Le Lan for productive discussions.

7. ETHICS STATEMENT

In this section we acknowledge (1) the broader ethical implications of generative music models in the context of our work, (2) the ethical implications of using crowdworkers to perform our subjective evaluation, and (3) our positionality as authors of this work.

7.1 Broader Ethical Implications

A recent work on the ethical implications of generative audio models [45] identified a set of potential harms specific to generative music models: (1) loss of agency and authorship, (2) stifling of creativity, (3) predominance of western bias, (4) cultural appropriation, (5) copyright infringement, and (6) climate impact of these models; we address this work with regard to each of these six harms.

This work is intended to provide creators with the ability to turn any sound gesture into a drum beat with their desired timbre. We see this as a means to provide music creators with additional agency; however, we do acknowledge that there is the (1) potential for removing agency or (2) stifling the creativity of percussion composition and production.

We recognize that our work is trained on a small dataset of drums and thus performs best with timbres present in that dataset, and so (3) may perform poorly with out-of-domain timbres such as traditional eastern music percussion instruments. This is a limitation of the dataset and current iteration of TRIA but not the proposed method itself, as future work could train TRIA on non-western drum beats to overcome this limitation.

In its current iteration, we do not believe there is a strong potential for (4) cultural appropriation with TRIA; however, if someone were to re-train TRIA on a dataset of percussion from a culture to which they do not belong, it would enable that act. In regard to (5) potential for copyright infringement, TRIA was trained on MusDBHQ-18 [1], which is licensed for any educational purposes. If TRIA were to be used for commercial purposes, it would require re-training on proprietary datasets or otherwise non-copyrighted work in order to protect the copyright holders of these tracks, though we are not proposing this work be used for commercial purposes.

Finally, we acknowledge (6) all generative models have an environmental impact—for transparency as encouraged by [46], we documented our computational resources used for training, training time, and number of parameters, which in all cases are far less than needed for competing models such as MelodyFlow. Based on our $4 \times$ NVIDIA A10G GPUs (150W) and 27 hours of training time, we estimate each training run has an energy cost of 16.2 kWh. For comparison, MelodyFlow was trained on $8 \times$ H100 96GB GPUs (350W), with no reported training time. If we

assume conservatively an equal training time of 27 hours, then one MelodyFlow training run would cost at least 280 kWh, or at a minimum $17 \times$ the energy consumption of TRIA.

7.2 Crowdworkers

Our subjective evaluation utilizing human listeners was approved (and determined to be exempt) under Institutional Review Board at the host university of the first author. We also ensured that each evaluator was paid a fair wage with an estimated hourly pay of \$18.75, which is above the minimum wage for every city in the United States. We also paid those who failed the listening test and thus could not partake in our study \$0.50 for their time. We used crowdworkers for this evaluation, and acknowledge that ethical use of crowdworkers goes beyond fair pay [47]; we tested the study among the author team prior to launch to ensure there would be no burden to workers beyond potential boredom and made sure the evaluators knew they could stop the study at any time.

7.3 Positionality

Finally, we would like to address the positionality of the authors. This is a diverse team of researchers, though we are predominantly from western developed countries (with one author being from the Global South). We are all both musicians and AI researchers, and thus share a mentality that AI technologies used for generative music can have a net positive impact as long as they are tools used to empower and assist musicians and creators rather than replace them. We acknowledge a bias in the conduct of this work reflecting an overall positive attitude towards AI technologies in this regard, and recognize that this is not a universal belief.

Ultimately, we believe that the benefits of this work far outweigh these potential risks, and we took care to keep them in mind as we conducted this research.

8. REFERENCES

- [1] Z. Rafii, A. Liutkus, F.-R. Stöter, S. I. Mimalakis, and R. Bittner, “The MUSDB18 corpus for music separation,” Dec. 2017. [Online]. Available: <https://doi.org/10.5281/zenodo.1117372>
- [2] G. L. Lan, B. Shi, Z. Ni, S. Srinivasan, A. Kumar, B. Ellis, D. Kant, V. Nagaraja, E. Chang, W.-N. Hsu, Y. Shi, and V. Chandra, “High fidelity text-guided music editing via single-stage flow matching,” *arXiv preprint arXiv:2407.03648*, 2024.
- [3] J. Gillick, A. Roberts, J. Engel, D. Eck, and D. Bamman, “Learning to groove with inverse sequence transformations,” in *ICML*, 2019.
- [4] A. Caillon and P. Esling, “Rave: A variational autoencoder for fast and high-quality neural audio synthesis,” *arXiv preprint arXiv:2111.05011*, 2021.

- [5] A. C. Santos and A. Cardoso, “From taps to drums: Audio-to-audio percussion style transfer,” in *Extended Abstracts for the Late-Breaking Demo Session of the 25th Int. Society for Music Information Retrieval Conf.*, 2023.
- [6] J. Engel, L. Hantrakul, C. Gu, and A. Roberts, “Ddsp: Differentiable digital signal processing,” *arXiv preprint arXiv:2001.04643*, 2020.
- [7] N. Demerlé, P. Esling, G. Doras, and D. Genova, “Combining audio control and style transfer using latent diffusion,” in *Proceedings of the 25th Int. Society for Music Information Retrieval Conf.*, 2024.
- [8] A. Ramires, R. Penha, and M. E. P. Davies, “User specific adaptation in automatic transcription of vocalised percussion,” *arXiv preprint arXiv:1811.02406*, 2018.
- [9] Vochlea, “Dubler 2.” [Online]. Available: <https://vochlea.com/products/dubler2>
- [10] O. Lartillot and F. Bruford, “Bistate reduction and comparison of drum patterns,” in *Proceedings of the 21st International Society for Music Information Retrieval Conference*. ISMIR, 2020.
- [11] B. Haki, B. Kotowski, C. Lee, and S. Jorda, “Taptamdrum: A dataset for dualized drum patterns,” in *Proceedings of the 24th International Society for Music Information Retrieval Conference*. ISMIR, 2023.
- [12] B. Kotowski, “Dualization of rhythm patterns,” Master’s thesis, Universitat Pompeu Fabra, 2020.
- [13] I.-C. Wei, C.-W. Wu, and L. Su, “Generating structured drum patterns using variational autoencoder and self-similarity matrix,” in *International Society for Music Information Retrieval Conference (ISMIR)*, 2019.
- [14] D. P. W. Ellis and J. Arroyo, “Eigenrhythms: Drum pattern basis sets for classification and generation,” in *International Society for Music Information Retrieval Conference (ISMIR)*, 2004.
- [15] D. Gómez-Marín, S. Jordà, and P. Herrera, “Network representations of drum sequences for classification and generation,” *Frontiers in Computer Science*, vol. 6, 2024.
- [16] G. Alain, M. Chevalier-Boisvert, F. Osterrath, and R. Piché-Taillefer, “Deepdrummer: Generating drum loops using deep learning and a human in the loop,” *arXiv preprint arXiv:2008.04391*, 2020.
- [17] S. Lattner and M. Grachten, “Drumnet: High-level control of drum track generation using learned patterns of rhythmic interaction,” in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019.
- [18] J. Nistal, S. Lattner, and G. Richard, “Drumgan: Synthesis of drum sounds with timbral feature conditioning using generative adversarial networks,” in *International Society for Music Information Retrieval Conference (ISMIR)*, 2020.
- [19] A. Lavault, A. Roebel, and M. Voiry, “Stylewavegan: Style-based synthesis of drum sounds with extensive controls using generative adversarial networks,” in *Proceedings of the Sound and Music Computing Conference (SMC)*, 2022.
- [20] J. Drysdale, M. Tomczak, and J. Hockman, “Style-based drum synthesis with gan inversion,” in *Extended Abstracts for the Late-Breaking Demo Session of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021.
- [21] H. Flores Garcia, P. Seetharaman, R. Kumar, and B. Pardo, “VampNet: Music generation via masked acoustic token modeling,” in *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2023.
- [22] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, “Roformer: Enhanced transformer with rotary position embedding,” *Neurocomputing*, vol. 568, no. C, Mar. 2024. [Online]. Available: <https://doi.org/10.1016/j.neucom.2023.127063>
- [23] R. Kumar, P. Seetharaman, I. K. Alejandro Luebs, and K. Kumar, “High-fidelity audio compression with improved rvqgan,” in *NeurIPS*, 2023.
- [24] Z. Borsos, M. Sharifi, D. Vincent, E. Kharitonov, N. Zeghidour, and M. Tagliasacchi, “Soundstorm: Efficient parallel audio generation,” *arXiv preprint arXiv:2305.09636*, 2023.
- [25] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman, “Maskgit: Masked generative image transformer,” in *CVPR*, 2022.
- [26] J. D. Parker, J. Spijkervet, K. Kosta, F. Yesiler, B. Kuznetsov, J.-C. Wang, M. Avent, J. Chen, and D. Le, “Stemgen: A music generation model that listens,” in *ICASSP*, 2024.
- [27] J. Ho and T. Salimans, “Classifier-free diffusion guidance,” in *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.
- [28] D. Garibi, O. Patashnik, A. Voynov, H. Averbuch-Elor, and D. Cohen-Or, “Real image inversion through iterative noising,” *arXiv preprint arXiv:2403.14602*, 2024.
- [29] A. Delgado, S. McDonald, N. Xu, and M. Sandler, “A new dataset for amateur vocal percussion analysis,” in *Proceedings of the 14th International Audio Mostly Conference: A Journey in Sound*, ser. AM ’19. New York, NY, USA: Association for Computing Machinery, 2019, p. 17–23. [Online]. Available: <https://doi.org/10.1145/3356590.3356844>

- [30] I. Pereira, F. Araújo, F. Korzeniowski, and R. Vogl, “Moisesdb: A dataset for source separation beyond 4-stems,” in *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2023.
- [31] OpenAI, “OpenAI GPT-4.5 System Card,” <https://openai.com/index/gpt-4-5-system-card/>, February 2025.
- [32] S. Ghosh, Z. Kong, S. Kumar, S. Sakshi, J. Kim, W. Ping, R. Valle, D. Manocha, and B. Catanzaro, “Audio flamingo 2: An audio-language model with long-audio understanding and expert reasoning abilities,” *arXiv preprint arXiv:2503.03983*, 2025.
- [33] M. Morrison, B. Tang, G. Tan, and B. Pardo, “Reproducible subjective evaluation,” in *ICLR Workshop on ML Evaluation Standards*, April 2022.
- [34] M. Cartwright, B. Pardo, G. J. Mysore, and M. Hoffman, “Fast and easy crowdsourced perceptual audio evaluation,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 619–623.
- [35] M. Cartwright, B. Pardo, and G. J. Mysore, “Crowdsourced pairwise-comparison for source separation evaluation,” in *2018 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE, 2018, pp. 606–610.
- [36] E. Rumbold, G. Tzanetakis, and B. Pardo, “Correlations between objective and subjective evaluations of music source separation,” 2024.
- [37] M. Zehren, M. Alunno, and P. Bientinesi, “High-quality and reproducible automatic drum transcription from crowdsourced data,” *Signals*, vol. 4, no. 4, pp. 768–787, 2023. [Online]. Available: <https://www.mdpi.com/2624-6120/4/4/42>
- [38] R. Vogl, M. Dorfer, and P. Knees, “Recurrent neural networks for drum transcription,” in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, 2016.
- [39] M. Heydari, F. Cwitkowitz, and Z. Duan, “Beatnet: Crnn and particle filtering for online joint beat down-beat and meter tracking,” in *Proceedings of the 22nd International Society for Music Information Retrieval Conference*, 2021.
- [40] Y. Chung, P. Eu, J. Lee, K. Choi, J. Nam, and B. S. Chon, “Kad: No more fad! an effective and efficient evaluation metric for audio generation,” *arXiv:2502.15602*, 2025. [Online]. Available: <https://arxiv.org/abs/2502.15602>
- [41] J. Nistal, S. Lattner, and G. Richard, “Comparing representations for audio synthesis using generative adversarial networks,” in *27th European Signal Processing Conference (EUSIPCO)*, 2019.
- [42] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharif, “Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms,” 2019.
- [43] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, Z. Yin, W. Wang, and M. D. Plumley, “Panns: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020.
- [44] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, “Largescale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.
- [45] J. Barnett, “The ethical implications of generative audio models: A systematic literature review,” in *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, 2023, pp. 146–161.
- [46] A. Holzapfel, A.-K. Kaila, and P. Jääskeläinen, “Green mir?: Investigating computational cost of recent music-ai research in ismir,” in *International Society for Music Information Retrieval Conference (ISMIR)*, 2024.
- [47] B. Shmueli, J. Fell, S. Ray, and L.-W. Ku, “Beyond fair pay: Ethical implications of nlp crowdsourcing,” *arXiv preprint arXiv:2104.10097*, 2021.