dB: A Web-based Drummer Bot for Finger-Tapping

Çağrı Erdem
Department of Informatics
University of Oslo
Oslo, Norway
cagrie@ifi.uio.no

Carsten Griwodz
Department of Informatics
University of Oslo
Oslo, Norway
griff@ifi.uio.no

ABSTRACT

dB is a web-based interface that serves as a "drummer bot" for exploring interactive groove-making experiences with an AI percussion system. This system, leveraging Variational Autoencoders (VAEs), transforms simple rhythmic inputs into complex drum patterns with microtiming and dynamics. Designed for accessibility and playfulness, dB is easily operated via a computer keyboard, making it suitable for a wide range of users. This paper outlines dB's foundational concepts, data collection, and a comprehensive overview of system and interface architecture. We then present our preliminary user study that investigated specific aspects of user engagement, including joy and boredom states, as well as perceptions of effort and control. The study's results underscore the musical background, expertise, and generational differences as significant influences on user experiences. Notably, test conditions characterized by greater randomness and rhythmic variation were consistently perceived as more engaging, and emerging trends were observed in user responses diverging over time.

Author Keywords

Human-AI collaboration, rhythm pattern generation, variational autoencoder, generative models, user studies

CCS Concepts

•Applied computing → Sound and music computing; •Computing methodologies → Machine learning; •Human-centered computing → Web-based interaction;

1. INTRODUCTION

Our body shapes our experiences [16]. Sound, the tactile sensation of a button click, or the sight of a dancer in motion all resonate through our physical and physiological responses. Using the human body as part of the musical instrument has been focal in new interfaces for musical expression (NIMEs) [39], with a variety of artificial intelligence (AI) techniques being explored for action–sound mappings since the early 1990s [27] in addition to the growing interest in interactive multi-agent systems [5, 40].

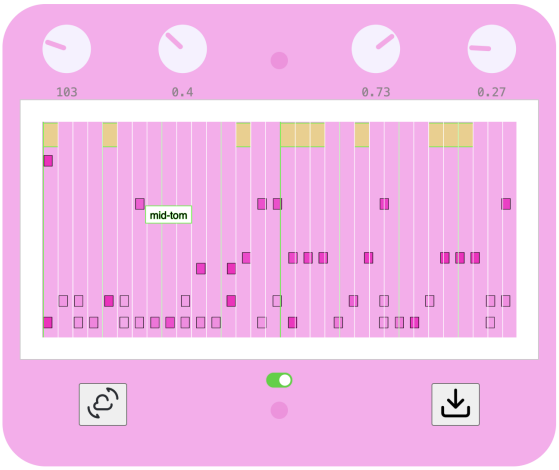


Figure 1: *dB* GUI features top-placed tempo, volume, and model knobs. At the bottom are the re-generation trigger and download buttons, and a piano roll is in the middle.

Yet, many studies in AI research today, including the music generation domain, tend to presume a separation between the body and the brain, often overlooking the significance of embodied interaction methods [9], which is particularly evident outside the NIME and interactive arts niches.

Therein lies a gap in comprehending what commonly engages users in interactions with AI music generation systems and what factors contribute to disengagement or boredom. Addressing this gap calls for a broader and more inclusive exploration of user studies, aiming to employ systems that resonate with a wider and more diverse audience.

To that aim, we designed *dB* to generate grooves via finger-tapping. We harness the fundamental appeal of rhythm and groove [45] as foundational in our study, transcending specific musical genres and styles, and emphasize the simplicity of finger-tapping as an approach that enables participation without prior musical knowledge. Here, the main question we embark on is:

- How do users interact and perceive through finger-tapping with a drummer bot in the browser?

This paper begins by outlining the background and key concepts that guided the development. We then detail the design principles and introduce the data and system components of the current prototype. Finally, we share insights from a preliminary study that explored the impact of varying randomness and rhythmic density conditions on user experiences.

NIME'24, 4–6 September, Utrecht, The Netherlands.

2. BACKGROUND

Central to all forms of *musicking* are the human body's roles in listening, playing, or dancing [37, 22]. Building on this foundation, our previous research has explored the capacity of musical AI to interact with human performers' bodily processes in real-time. This approach, termed "embodied perspectives" by the first author [10], involved utilizing bodily signals, such as changes in muscle activation, acceleration, and orientation during sound production.

In one such study, we analyzed guitarists' physiological processes in the forearm muscles for predictive modeling of action–sound relationships and developed "air instruments" [11]. Another project trained intelligent agents to function as musical partners, as in *CAVI*, where an AI model interpreted the performer's movements to generate control signals for live sound processing [13].

Professional musicians who performed live with CAVI consistently noted the need for the system to evolve beyond just tonal and rhythmic accuracy, especially in improvisational contexts. They stressed the significance of AI understanding the meanings of gestures to establish a communicative bond with human musicians [12]. This requires interpreting musical decisions alongside higher-level human body motion aspects like mood changes or emotions expressed through facial expressions and physical movements.

Recognizing the importance of affective elements in musical communication, our research shifts focus toward understanding the nuanced expressions of user experiences, with the goal of integrating them as part of the future systems we develop. In this transition, we start from an interaction scenario as constrained as finger-tapping, which acts as a bottleneck between the human body and music-making. This deliberate choice allows us to explore in what ways such minimal bodily actions can shape how users "feel" in creating grooves. By targeting a broader user group beyond musicians and interactive arts enthusiasts, we aim to identify the differences and boundaries within and outside these specialized niches.

2.1 Musical Rhythm Generation

AI techniques have extensively been applied to the broad domain of audio and music generation [23, 48]. Notable ones that focused on rhythm and groove are GrooVAE's sequence-to-sequence VAE network for drum patterns [17], which is also the main inspiration for the presented work, in addition to the "AI drummer" employing temporal convolutional networks for human improvisation response [30], and the study of microtiming in Brazilian percussion [47].

Other innovative approaches include intelligent agents that can learn drumming from human movements [41], conditioning LSTMs for rhythm composition [29], generating EDM rhythms with GANs [43], or exploring Transformer neural networks for rhythmic pattern generation [32, 19].

Our project is informed and inspired by these diverse AI music systems but diverges in its emphasis on developing an accessible and engaging web application, hinging on three core interdisciplinary concepts: Groove, playfulness, and accessibility. The accessibility aspect relies on ease of use, while playfulness is cultivated by combining groove-making with varying levels of randomness and rhythmic density akin to the model's generative parameters.

2.2 Groove

In music psychology, *groove* is defined as an instinctive urge to move in sync with a musical rhythm [46], hence fundamentally involving the body, moods, and emotions. Our natural tendency for synchronization and rhythmic action–perception coordination are important themes in music and cognition research [26, 33]. The cross-cultural potential of rhythm and groove makes it an ideal focus for our research objectives of reaching diverse users. Moreover, the symbolic data format's efficiency in handling rhythmic and groove-based information complements our approach.

2.3 Playfulness

The concept of *playfulness* puts an emphasis on surprise and uncertainty over rules and conventions [28]. This approach, akin to children's "deliberate play," is intrinsically rewarding and considered foundational for specialization and expertise [6]. Playfulness is also an important theme in developing NIMEs [1, 42]. In our system, this translates to an environment where users can easily explore a musical groove-creation process that can be predictable and surprising at the same time, a balance often suggested as a sweet spot for groove experiences [38].

2.4 Accessibility

Accessibility is a key consideration among NIME practitioners, often described as having a "low entry fee" [44], to make NIMEs usable by individuals with limited or no musical background [31, 15, 14]. Our design of dB reflects this philosophy, aiming to be accessible to a wide audience, regardless of their skill or affinity. dB's browser-based operation simplifies the user experience, eliminating the need for additional setups. Additionally, our choice of data, featuring straightforward groove patterns, was deliberate to facilitate the connection with and understanding of the generated rhythms.

3. DATA AND REPRESENTATION

For AI research, the availability of annotated datasets plays a crucial role. Notable examples in the realm of symbolic rhythm and beat generation include the *Magenta Groove MIDI Dataset* (GMD), which encompasses nearly 14 hours of expressive drumming [17], and its *Expanded* version [3]. Another one is the *TapTamDrum* dataset of 345 patterns played by skilled drummers [18]. These datasets are instrumental for research and maintaining objectivity.

However, the creative potential that lies in handpicking and curating a dataset from the ground up represents an often overlooked aspect. Drawing inspiration from the compositional use of data [13], our data collection included carefully listening to drum grooves, considering them at different tempos, and selecting each sample individually. We explored a variety of royalty-free MIDI drum groove collections, ranging from large libraries with hundreds of samples to single files uploaded by individuals to online resources. Due to the sheer number and variety of these resources, listing each one is not feasible.

3.1 Curation

This project focused on eight-note beats commonly found in rock and heavy metal. We restricted our dataset to grooves in a 4/4 meter, avoiding shuffle or swing feels, and selected more regular beats with minimal syncopation. This was done to enable typical physical responses to music, such as head-banging or body-swaying, often associated with entrainment. That is, the phenomenon of biological or mechanical systems rhythmically synchronizing, such as people clapping in unison [4], and is considered a predictor of joy and affective experiences [34].

3.2 Preprocessing

The preprocessing steps involved addressing the differences in each MIDI file, such as varying lengths, number of tracks and drum parts, and pulses per quarter note (PPQN) resolution that denotes the number of pulses or ticks in a quarter note. Upon the exclusion of fills and grooves with non-4/4 time signatures, we tailored the metadata by setting a consistent tempo, consolidating the note events into one track and a fixed nine-part drum set, and standardizing the PPQN to 480. This process yielded approximately 13,000 two-bar data segments, ready for further analysis and model training.

3.3 Representation

We transform MIDI files into structured data representations, drawing on methods similar to those described in [17]. Each MIDI file is processed into four arrays: Hits (H), Velocities (V), Time Offsets (O), and Metadata (M). The H array is binary, indicating note hits. V details the dynamics of drum hits, and O captures microtiming deviations with respect to PPQN, with both normalized within the ranges of $[0, 1]$ and $[-1, 1]$, respectively. Metadata (M) includes tempo and time signature information for model conditioning. This results in a comprehensive representation of each MIDI drum groove across two 4/4 bars, quantized to sixteenth notes.

We utilize two key functions in this transformation: f_{map} for disaggregating MIDI data into arrays H, V, and O, and $f_{compress}$ for reducing H into a compact, one-dimensional array H_t, as per the "Pattern Category" concept [35]. This concept divides drum grooves into three rhythmic layers: *Downbeat*, *Backbeat*, and *Pulse*. We focus on the Pulse ($\mathcal{P}$) layer, which typically includes hi-hats and ride cymbal notes. We suggest that $\mathcal{P}$ intuitively resonates with finger-tapping as there can be a natural correspondence between the movement patterns in tapping with a finger and playing cymbals with a stick. We aim to capture the full drum groove's essence in a singular dimension by processing this layer into H_t.

$$H, V, O = f_{map}(MIDI, D) \quad (1)$$

$$H_t = f_{compress}(H, \mathcal{P}) \quad (2)$$

The final data representation comprises H, V, O, M, and H_t to combine the learned insights from multi-dimensional arrays with an intuitive grasp of the rhythm offered by the one-dimensional H_t array.

4. SYSTEM OVERVIEW

4.1 Model Architecture

At the core of our Variational Autoencoder (VAE) model's architecture is the processing of H_t arrays, which are quantized representations of finger-tapped rhythms. These H_t inputs capture only the rhythmic "hits" without any dynamics or microtiming information. Once processed through the model, these H_t inputs are transformed into full drum set patterns encompassing Hits (H'), Time Offsets (O'), and Velocities (V') for all nine predefined parts of a drum set. In the following, we briefly introduce the main components of the model architecture as depicted in Figure 2.

4.1.1 Encoder

The encoder in our VAE architecture utilizes a bidirectional long short-term memory (BiLSTM) layer with 512 units designed to process the temporal patterns inherent in the input H_t matrices, each comprising 32 timesteps. The input of the BiLSTM is concatenated with Metadata (M), specifically incorporating tempo information, to condition the latent space parameters as in [36]. This methodology ensures that the metadata influences the model's output, allowing the generation of tempo-aware rhythm sequences.

4.1.2 Latent Space

The encoder's BiLSTM processes the H_t input and outputs the conditioned latent space parameters: the mean (Z_μ) and the variance (Z_σ). A dense layer then processes the latent vector Z, which is sampled through reparameterization, introducing controlled randomness into the model's generative process:

$$Z = Z_\mu + Z_\sigma \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I). \quad (3)$$

The sampled latent vector Z is subsequently concatenated with the original H_t and M inputs. This technique of "skipping" connections as in [21] ensures that both the tapping and meta information are preserved and condition the decoding process, facilitating a more accurate reconstruction of the rhythm patterns.

4.1.3 Decoder

The decoder reconstructs the rhythmic patterns from the latent space representation. It utilizes a series of two unidirectional LSTM layers as in [36], each followed by a dropout layer to prevent overfitting. At the last stage, time-distributed dense layers with specific activation functions –*softmax* for Hits, *sigmoid* for Velocities, and *tanh* for Time Offsets– shape the outputs to the expected dimensions.

4.2 Training

The training process is auto-regressive to anticipate future steps in a sequence. For better model selection, we used K-Fold cross-validation [20], evaluating data across various segments. Here, only the "tapping" matrices (H_t) are passed to the Encoder LSTM, then conditioned with M). The Decoder's outputs (H', V',O') are then compared with the original (H, V, O) matrices to compute the loss.

The model is trained using a composite loss function, $\mathcal{L}$, which combines *categorical cross-entropy* for the Hits with *mean squared error* for the Velocities and Time Offsets:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}(H, \hat{H}) + \lambda_V \cdot \mathcal{L}_{\text{MSE}}(V, \hat{V}) + \lambda_O \cdot \mathcal{L}_{\text{MSE}}(O, \hat{O}), \quad (4)$$

where $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss, $\mathcal{L}_{\text{MSE}}$ is the mean squared error loss, and λ_V and λ_O are the weighting terms for the respective losses.

Additionally, a Kullback–Leibler (KL) divergence loss is included to regularize the latent space by penalizing deviations from the assumed standard normal distribution:

$$\mathcal{L}_{\text{KL}} = -\frac{1}{2} \sum (1 + \log(\sigma^2) - \mu^2 - \sigma^2). \quad (5)$$

The KL loss, scaled by a factor β, balances reconstruction fidelity with latent space regularization. We train the network using this overall loss function together with the Adam optimizer [24].

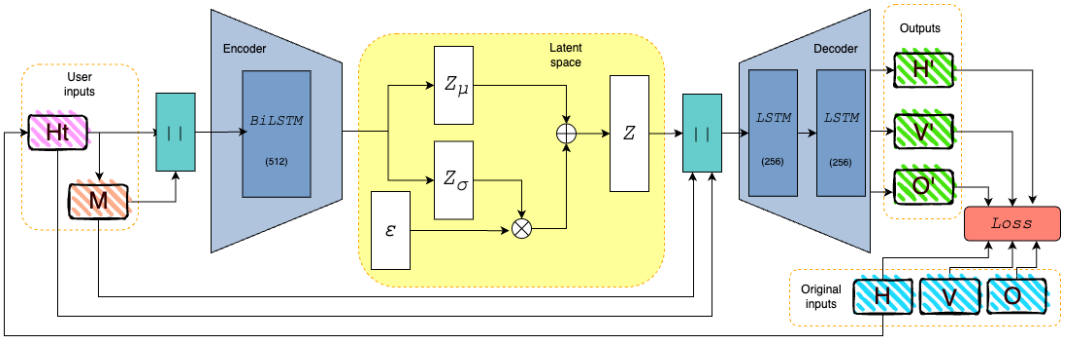


Figure 2: A simplified diagram of dB's model architecture. Tapping inputs (H_t) that are derived from the *pulse layer* of the original drum grooves (H) are fed into the encoder, and metadata (M) is utilized to condition the model. The encoder produces the parameters Z_μ and Z_σ for the latent space, where a random variable ϵ is introduced to generate the latent variable z through reparameterization. This latent variable is then concatenated with the encoder's outputs to enhance the decoding process. The decoder generates the predicted outputs H', V', and O, and the main loss is computed by comparing these with their true counterparts.

4.3 Sampling and Generation

The trained model generates new outputs as H', V', and O' arrays derived from the finger-tapped H_t matrices. In addition to passing the tapping inputs to the model, the browser interface offers knobs and buttons to control tempo, sampling, and thresholding parameters, all contributing to the conditioning of the inference. In the following, we discuss the core methods of our generative processes.

4.3.1 Latent Space Sampling

The encoder computes two key vectors for any given input: the mean vector μ and the standard deviation vector σ. A point z from the latent space is then sampled using the following equation:

$$z = \mu + \epsilon \cdot \sigma, \tag{6}$$

where ϵ is a random noise vector drawn from a standard normal distribution. This sampling method introduces controlled randomness into the generation process, where the magnitude and direction are dictated by ϵ.

4.3.2 Probability Distribution Shaping

We utilize the softmax function to shape the probability distribution of the outputs. This function converts a vector of raw scores s into a probability distribution P:

$$P = \text{Softmax}(s_i) = \frac{e^{s_i/T}}{\sum_j e^{s_j/T}}, \tag{7}$$

The temperature T acts as a scaling factor to the logits prior to the softmax function. A higher temperature (e.g., $T > 1$) increases the randomness of the outputs by flattening the probability distribution. On the other hand, a lower temperature (e.g., $T < 1$) results in more deterministic outputs, sharpening the distribution and thus yielding higher model confidence levels.

4.3.3 Output Manipulation

Additionally, dB incorporates three generative techniques for manipulating the latent space. The first technique introduces controlled randomness to the latent vector, enriching the variety of the outputs. This process can be represented as $z' = z + \mathcal{N}(0, \sigma^2)$, where z is the original latent vector and $\mathcal{N}(0, \sigma^2)$ denotes the Gaussian noise.

The second technique explores the neighboring regions of the latent space for smoother transitions, adjusting the vector randomly in those regions. This is expressed as $z' = z + \delta \cdot \text{direction}$, with δ representing the exploration factor.

Lastly, the system employs a blending method between current and previous latent points to ensure smooth transitions and maintain coherence in the groove sequences generated subsequently. This interpolation is defined as $z' = \alpha z_1 + (1 - \alpha) z_2$, where z_1 and z_2 are latent vectors being interpolated, and α is the blending factor.

4.3.4 Dynamic Thresholding

In conjunction with sampling and generative techniques, dynamic thresholding plays a critical role in terms of the groove's rhythmic density. This user-controlled parameter adjusts the threshold level applied to the output probabilities. Here, the user determines how the generated softmax probabilities are translated into actual note-on events in the H' matrix. Thus, a lower threshold allows for the inclusion of less likely drum hits, potentially with more rhythmic density. Or, higher temperatures can yield outputs with lower confidence levels. Such outputs might result in sparse or even no generation without dynamic thresholding when, for example, a high threshold is used. Therefore, a careful calibration of the temperature and threshold is essential to balance the novelty of generated rhythms with the model's confidence in its predictions.

4.4 User Interface

The user interface is based on a client-server architecture, where the front-end application interacts with the back-end server that executes the trained model and data processing.

4.4.1 Back-end

The back-end, implemented in Python and TensorFlow, is responsible for passing the tapping inputs and generation parameters to the trained model, post-processing the generated outputs, and sending them back to the client. Once the generation is complete, the output that comprises H', V', and O' matrices is converted into a MIDI object using the Mido MIDI library.[1] This MIDI object is then encoded into JSON-formatted MIDI bytes and sent back to the front-end.

[1] `https://github.com/mido`

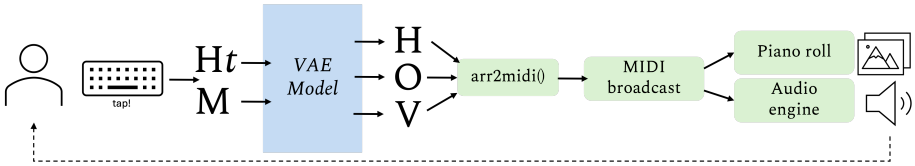


Figure 3: An overview of dB's signal flow. The user inputs finger-tapped rhythmic patterns. The VAE model generates drum grooves broadcasted as MIDI messages for playback and visualized as a piano roll.

4.4.2 *Front-end*

Our front-end application, developed with HTML, CSS, and JavaScript, offers a user-friendly interface for rhythm input and parameter adjustment via a computer keyboard. It combines the functionalities of handling interactive widgets, pre-processing of finger-tapped rhythms, communicating them with the back-end via Ajax requests, parsing & broadcasting the fetched MIDI data, and data visualization. The audio engine, responsible for sonifying tapped rhythms, metronome ticks, and drum audio sampling, is integrated within these modules. Web Workers are utilized to manage server requests and handle the asynchronous fetching of data, MIDI queuing, and clock synchronization.

4.4.3 *Interaction Design*

The interactive controls include knobs to adjust metronome BPM, click volume, sampling temperature, and threshold settings (Figure 1). At the center lies a piano roll, created with p5.js[2] to visualize the finger-tapped inputs and generated drum grooves. The lower section houses buttons for initiating the generation of new outputs based on the last tapping inputs and MIDI file download options.

As shown in Figure 3, user interactions with dB involve tapping rhythms using the space bar and storing them with the 'A' key on the keyboard. Accompanied by blinking lights and sounds, this setup provides clear feedback. Users can adjust the stochasticity of outputs through temperature and threshold knobs, balancing control, surprise, and rhythmic complexity. Similar to our constrained "bottleneck" interaction strategy based on finger-tapping, this approach aims for a more concentrated creative exploration.

4.4.4 *Audio Output*

The audio output functionality is composed of:

1. A broadcasting script that employs the Web Audio and Web MIDI[3] APIs for MIDI parsing, clocking, note scheduling, and looping.

2. A queuing system based on Web Workers, which orchestrates the sequence of operations from receiving tapped rhythms to fetching back the generated MIDI bytes and timing the playback.

3. Drum sampler and audio effects (EFX) scripts built with Tone.js,[4] serving as a virtual percussion instrument. This allows for flexibility in audio playback, including the option to route MIDI events to external instruments.

4.4.5 *Deployment*

The dB application is hosted on a virtual machine under the Norwegian Research and Education Cloud (NREC) scheme. It employs Gunicorn[5] and Nginx[6] web servers for deployment. As the application server, the former manages and executes application logic, while the latter serves as the front-facing server, handling static content and directing dynamic requests to Gunicorn.

4.4.6 *Error handling*

We incorporated safeguards in our system for MIDI parsing, data integrity, and client-server communication. These include checks for negative MIDI delta values to prevent conversion errors, mechanisms to handle potential empty arrays generated due to specific sampling and threshold values, and adaptations for different user inputs, such as adjusting for single taps or truncating lengthy sequences.

5. EXPERIMENTS

Following the implementation of dB's functional prototype, a preliminary study was conducted with 63 participants to probe their experiences in interactive groove generation. This study primarily investigated boredom, good feeling, control, and tiredness experiences under varying tempo, randomness, and rhythmic density conditions.

5.1 Setup

The user study interface (Figure 4) was a simplified version of the original design (Figure 1). This simplification aimed for participants to focus on rhythmic play and improvisation in a controlled environment with pre-determined parameter combinations for each test condition.

5.2 Procedure

We used the dB system within a master's course at the Department of Informatics, University of Oslo. Each student conducted the study using a consistent framework and reported findings, while the students themselves did not act as participants, nor was participation part of mandatory coursework. This method engaged a diverse participant pool from the social spheres of conductors.

The 30-minute study included an introduction, an initial questionnaire, a practice session, and nine randomized 2-minute dB interaction sessions, each followed by a survey, and concluded with a final questionnaire. Participants inputted three tapped rhythms in each session under predefined parameter conditions detailed in appendices (A).

The conductors followed a detailed protocol for collecting consent forms and gathering the data. Responses were measured on a 5-point Likert scale ranging from "strongly disagree" to 'agree strongly," and "none" to "very much." Data from 11 participants were discarded due to inconsistencies, leaving 468 analyzed interaction tasks from 52 individuals ($\mu_{\text{age}} = 29.79$ [15, 74] $SD = 12.50$, 19 females, 1 nonbinary, 32 males).

[2] `https://p5js.org/`
[3] `https://github.com/djipco/webmidi`
[4] `https://tonejs.github.io/`
[5] `https://gunicorn.org/`
[6] `https://www.nginx.com/`

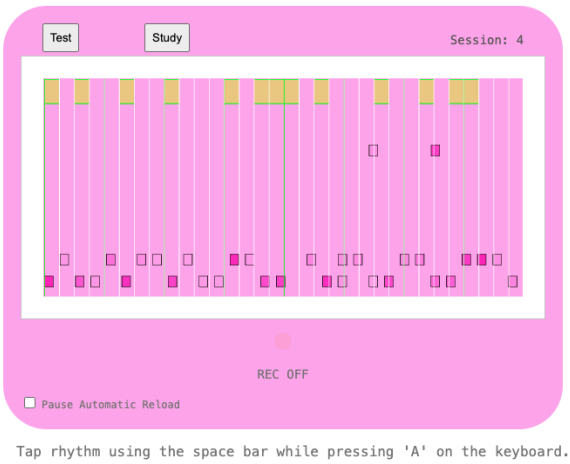


Figure 4: Simplified user study interface. In this version, all parameters are pre-determined except for the conductor controls: a *Study Button* for initiating new study conditions, a *Test Button* for system setup or testing, and a checkbox to *Pause* the auto-reload, set to a 2-minute interval, when more time was needed (e.g., for questionnaire completion).

6. RESULTS

Acting as meta-conductors, we further analyzed the data, extracted the trends and statistically significant results, and synthesized them under four emerging areas: How a user's musical affinity affects their interactions with the system, how the control dynamics and previous NIME experience play a role, how different sessions' unique parameter combinations of temperature, threshold, and tempo influence the user experiences, and how age impacts the engagement.

We started our investigation with how musical inclination affects user engagement with the dB system, dividing participants into two archetypical groups: *Musicking* (23 student or semi-professional musicians and avid music listeners) and *Non-musicking* (29 averagely interested in music and not particularly interested in music). These categories initiated the exploration in the following sections.

6.1 User Engagement and Musical Affinity

We scrutinized participant responses to four principal statements —"I felt bored," "I felt good," "It was tiresome," and "I felt inspired"— which were evaluated after each session. Following significant deviations from normality as indicated by the Shapiro-Wilk test, we investigated these specific user experiences in connection to musical affinity and background through non-parametric Kruskal-Wallis H tests.

A direct comparison between the archetypical groups initially did not reveal significant differences. We then refined our analysis by specifically considering participants who have (1) some familiarity with improvisational music, (2) some experience with NIMEs and interactive systems, and (3) who reported exerted effort levels at or above average and (4) higher ratings for the system's influence on their musical choices.

This targeted approach led to significant differences. The Non-Musicking sub-groups generally reported more positive experiences, feeling more content, less bored, less fatigued, and more inspired. The exceptions were the "It was tiresome" ratings based on NIME experience ($p = 0.054$) and the "I felt good" ratings based on the exerted effort ($p = 0.102$). Details of these findings are displayed in Figure 5.

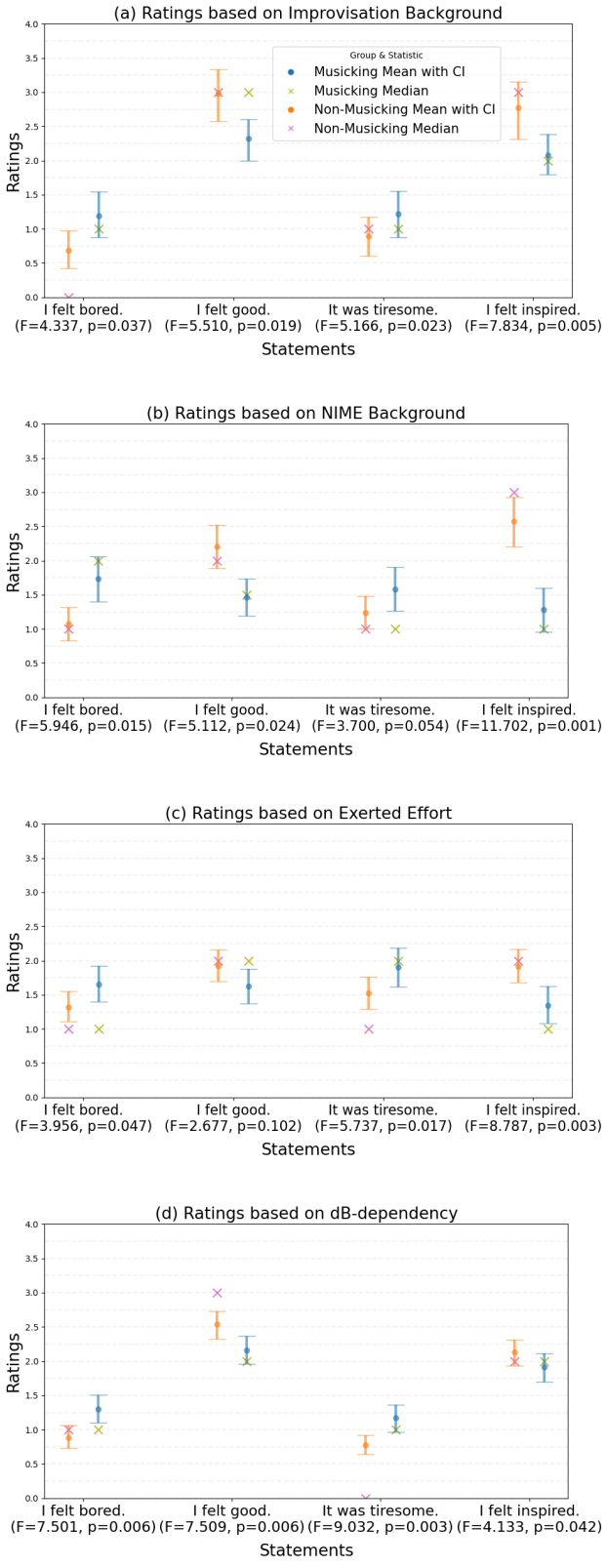


Figure 5: Comparison of average ratings between Musicking and Non-Musicking groups using medians (marked by 'x') and means (circles). Bootstrap confidence intervals (CI) illustrate the range of estimations and overlaps, reflecting the variability in responses. The Kruskal-Wallis test assesses the statistical significance of differences between groups for each statement, with F and p-values provided below the statements.

Furthermore, we performed a preliminary temporal analysis of responses by normalizing participants' total study durations using response timestamps and calculating average responses within fixed time windows. The regression analysis, as illustrated in Figure 6, uncovered patterns that indicated a decline in both groups' boredom levels towards the study's conclusion, while the Musicking group's fatigue trend increased, unlike the others.

6.2 NIME Familiarity and Control

We further narrowed our focus on the users' backgrounds and prior experiences, particularly knowing that individuals familiar with NIMEs and interactive systems tend to experience diverse control and interaction dynamics. Such individuals are typically more attuned to their influence on the output and often comfortable with waiving control to a generative system. Thus, participants were categorized as *no experience* ones who rated their experience with interactive musical interfaces as 0, and those with *prior experience* who had a rating of 1 or above in the pre-study questionnaire. Perceived control was derived from the responses to a question in the post-study survey, "Who had the overall control of the interaction?," with below-average ratings indicating that the participant *felt less control* and above-average indicating *felt more control*.

Table 1: Participant categories based on their self-assessed experience with NIMEs and their perception of control during the interaction.

	felt less control	**felt more control**
no experience	group 1 (15 participants)	group 2 (13 participants)
prior experience	group 3 (12 participants)	group 4 (12 participants)

To streamline the testing process, we pre-determined parameters (see Appendix A for the details of test conditions) across three discreet Tempo, Temperature, and Threshold levels, and we applied the Friedman test for non-parametric ANOVA by ranks, which is ideal for situations where multiple participants rate a series of conditions. Group 3, with the statement "I felt bored," displayed marked rank differences in comparison to other groups.

Post hoc Nemenyi test for comparisons of all sessions against each other determined significant variations in boredom levels between sessions $S_{1,mid}$ and $S_{3,mid}$, as well as session $S_{2,slow}$ and $S_{3,mid}$:

$$\text{"I felt bored" (group 3)} : \begin{cases} S_{1,mid} \text{ vs. } S_{3,mid},\ z = 4.690, p = 0.045 \\ S_{2,slow} \text{ vs. } S_{3,mid},\ z = 4.432, p = 0.003 \end{cases} \tag{8}$$

This suggests that Group 3 participants, with prior experience with NIMEs and interactive systems and who felt that dB had the overall control, were more attuned to changes in model generation parameters. They reported less boredom in sessions characterized by increased randomness (such as Session 8), as detailed in the session parameter combinations (A). Notably, when we substituted NIME familiarity with the improvisation experience, the pattern persisted, although only the boredom variance between Sessions 4 ($S_{2,slow}$) and 8 ($S_{3,mid}$) reached significance ($p = 0.0102$).

6.3 Generation Parameters and Their Effect

The results from the repeated measures ANOVA, with session identifiers as a key variable, indicated significant differences ($F = 3.133, p = 0.004$), specifically in the "I felt bored" statement across test conditions. The Greenhouse-Geisser correction was applied due to the violation of sphericity, as indicated by Mauchly's test. The correction factor ($\epsilon = 0.8139$) was used to adjust the degrees of freedom. The key findings of the post hoc analysis included significant differences in boredom levels, as shown in Table 2 and Figure 7.

Table 2: The post hoc Tukey HSD test results show that Sessions 8 ($S_{3,mid}$) and 9 ($S_{3,fast}$) received significantly lower boredom levels.

A	**B**	**Mean diff.**	**T-stat**	**p-value**
$S_{1,slow}$	$S_{3,mid}$	0.5	3.01	0.004
$S_{1,mid}$	$S_{3,mid}$	0.46	3.05	0.003
$S_{2,slow}$	$S_{3,mid}$	0.57	3.48	0.001
$S_{2,slow}$	$S_{3,fast}$	0.46	3.15	0.002
$S_{2,mid}$	$S_{3,mid}$	0.4	3.27	0.001
$S_{2,fast}$	$S_{3,mid}$	0.48	3.4	0.001

The results for other session comparisons did not reach the threshold for statistical significance after applying the Benjamini/Hochberg FDR correction adjustment for multiple comparisons. As for general trends, $S_{3,mid}$, followed by $S_{3,fast}$, was marked as the session with the most disagreement among participants regarding feeling bored –as opposed to session $S_{1,slow}$, $S_{1,mid}$ and $S_{2,slow}$– depicted in Figure 8. These results provide valuable insights into how different session conditions influenced participants' experiences of boredom.

6.4 The Age Factor

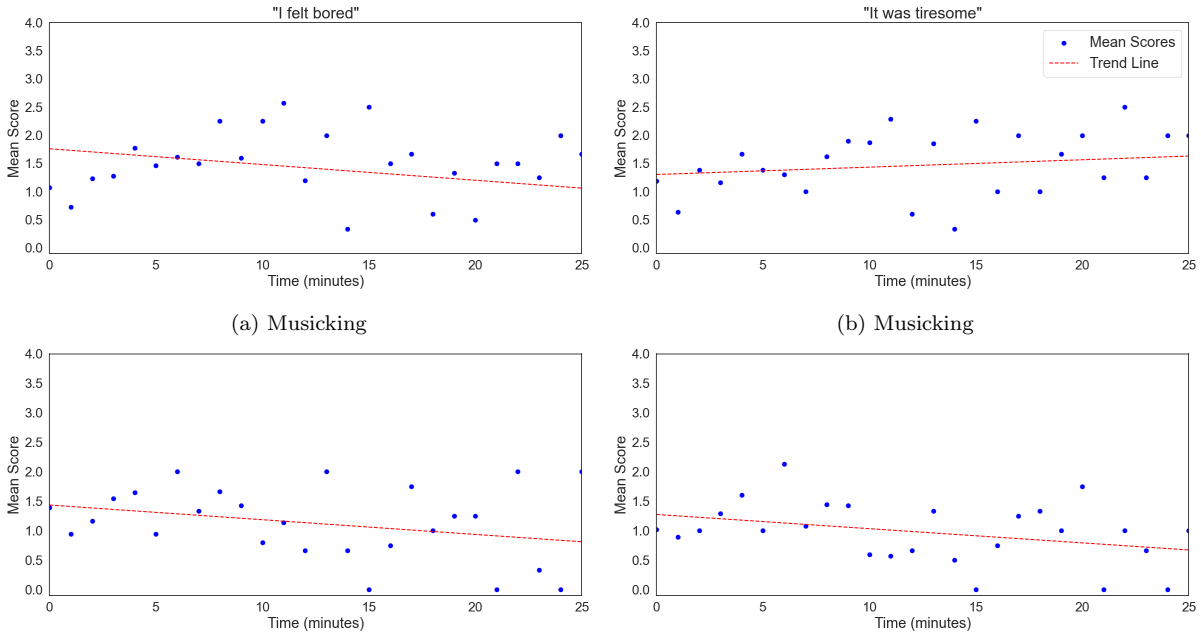
As for demographics, we observed distinctions in connection to the age group. Participants from the so-called *Generation Z* cohort (aged 15–26 years), born in 1997 and younger [8], varied in their responses compared to older participants (27–74 years). The Shapiro-Wilk test, applied to the "I felt bored," "It was tiresome," and "I felt good" responses, yielded Test Statistics as 0.835, 0.830, and 0.881, respectively (all $p < 0.001$), indicating significant deviation from normality in the data. This suggests age-related differences in interaction with music AI systems. Further analysis using linear mixed models (LMMs) regression confirmed the findings partly:

- **"I felt good"** ($\beta = 0.386$): "Positive feelings tend to increase with age" ($p = 0.003$)
- **"It was tiresome"** ($\beta = -0.160$): "Tiredness tend to decrease with age" ($p = 0.187$)
- **"I felt bored"** ($\beta = -0.615$): "Boredom levels tend to decrease with age" ($p < 0.001$)

These results imply that older users tend to feel better and less bored, while age may not significantly affect the users' tiredness.

7. DISCUSSION

We grouped participants based on their musical affinity, reflecting the concept of *musicking* [37]. This archetypical grouping did not significantly differentiate their experiences of enjoyment, fatigue, inspiration, or boredom. However, further segmentation based on participants' prior knowledge and specific aspects of their physical and sensory involvement with the system revealed nuanced patterns in user responses (Figure 5).



(a) Musicking

(b) Musicking

(c) Non-musicking

(d) Non-musicking

Figure 6: Dots represent the mean scores of each group's ratings over the course of the study, and the linear regression line indicates the trend.

The Non-musicking group generally reported more positive experiences, indicating dB's accessibility. Yet, closer inspection across different sub-groups showed that higher effort levels and a perceived lack of control uniformly affected user engagement, regardless of musical affinity. This points to an embodied perspective in user interactions.

Notably, participants with prior NIME experience registered higher inspiration levels, especially those who identified as Non-musicking. This suggests that individuals with limited musical interest might engage with interactive music systems, potentially drawn by technological or playful aspects. Participants familiar with improvisation consistently reported higher positive feelings, indicating that such familiarity can contribute to engagement with novel interactive experiences.

We focused further on the statement "I felt bored," as it frequently showed significant variance. Participants with NIME experience who perceived dB as more controlling reported less boredom in sessions with higher randomness and rhythmic density. This hints that engagement is influenced by the comfort in sharing control and interaction agency.

When we examined the divergence in responses over time, we observed that participants tend to become less bored, suggesting a possible mastery growth (Figure 6). The Musicking group, however, displayed increased fatigue, potentially due to the system becoming too monotonous or uncontrollable. Despite this, they felt more room for exploration than Non-musicking participants (Figure 9b).

Comparing session means in Table 2 revealed that sessions $S_{3,mid}$ and $S_{3,fast}$ were less boring that sessions $S_{1,slow}$, $S_{1,mid}$, $S_{2,slow}$, $S_{2,mid}$ and $2_{2,fast}$ across all participants. This pattern, as illustrated in Figure 7, indicates that randomness, rhythmic variation, and medium to faster tempos are less likely to bore participants. This highlights surprise as a critical aspect of a playful experience, stressing some recent findings on what makes interactive arts engaging [25].

Furthermore, Gen Z participants experienced more boredom and less positive feelings compared to older ones. While this relates to broader trends in media consumption and attention spans among younger generations, it is beyond our scope yet useful for future research and designing age-specific user experiences.

Key limitations in our study included the use of a computer keyboard for interaction, which could impact the sense of control due to variable build quality and responsiveness. One participant noted dB's lack of "high-resolution," highlighting this issue together with the limited quantization of finger-tapped rhythms. This echoes the difference between interaction and influence, where the latter rather has longer-term effects without explicit, direct causality [2].

Another key limitation, also a critical challenge in music AI, lies in the lack of perceptual validation methods for loss and accuracy. Our initial method for translating drum grooves into finger tapping ([35]) led to better results than, for example, compressing all drum layers into one dimension. Still, it inevitably calls for further study to model the congruity between physical actions and musical rhythms.

Notably, a considerable proportion of participants did not report "feeling skillful," as seen in Figure 9a. This might relate to an insufficient sense of ownership, highlighting the need for a better balance between surprise and controllability in crafting playful experiences. Nevertheless, most users found dB's output coherent and the interaction engaging overall, as indicated in Figure 9c.

8. CONCLUSION

This paper introduced a web-based drummer bot, *dB*, and explored interactive groove-making experiences with it. We presented the system components, setting the stage for our preliminary user study. The results highlighted how musical background and personal experiences can significantly shape users' interactive experiences. Intriguingly, we found that higher randomness and rhythmic variation tended to be less boring, indicating a connection between complexity and user engagement. These findings provide valuable insights for future efforts aimed at creating more inclusive and engaging experiences with AI music systems.

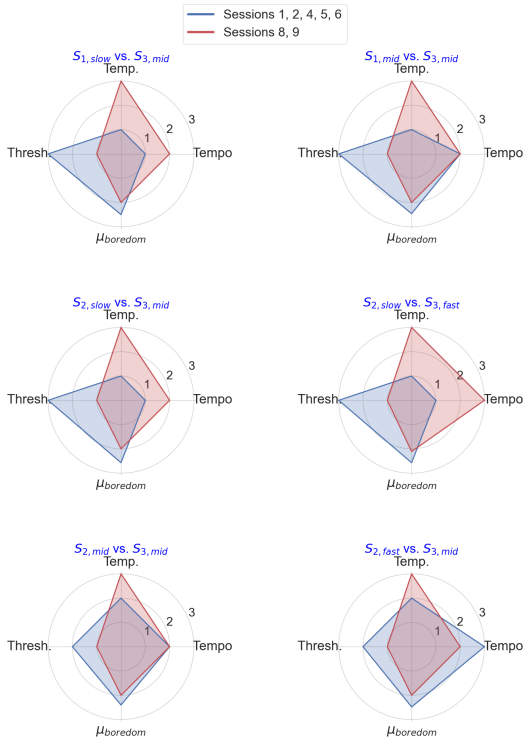


Figure 7: Charts showing Tempo, Temperature, and Threshold combinations, all in the $1-3$ range, with the mean "I felt bored" ratings normalized to the same range. Sessions 8 ($S_{3,mid}$) and 9 ($S_{3,fast}$) parameters yielded significantly lower boredom means, while the parameter combinations of other sessions received higher boredom ratings on average. We observe that the combination of higher randomness (*Temp.*) and low rhythmic density (*Thresh.*) positively affected users' engagement.

dB can be accessed at `https://2groove.live/`. We shared the code and the curated dataset at a public repository at `https://github.com/cerdemo/2groove`, inspired by the Open Research practices. Committed to an iterative research and design process, we aim to conduct more targeted studies on musical human–AI interactions, drawing on the knowledge gained from this initial exploration.

9. ETHICAL STANDARDS

Our study was conducted according to the code established by the National Ethical Committee for Natural Science and Technology (NENT) [7]. Our master's students assisted the study by recruiting participants from their social networks, ensuring they were not enrolled in the relevant course and that participation was voluntary. The participants were informed about the research's purpose, guaranteed anonymity, and confirmed that no personal data was collected.

10. ACKNOWLEDGMENTS

We are grateful to the "student conductors" from the IN5060 course at the Department of Informatics, University of Oslo, and the participants they recruited. Special thanks to Sabry Razick and Ashen Wijesiri from the IT department (USIT) for their assistance with the virtual machine. This work was partly supported by the EU 6G SNS programme under grant agreement No. 101096452 (IMAGINE-B5G).

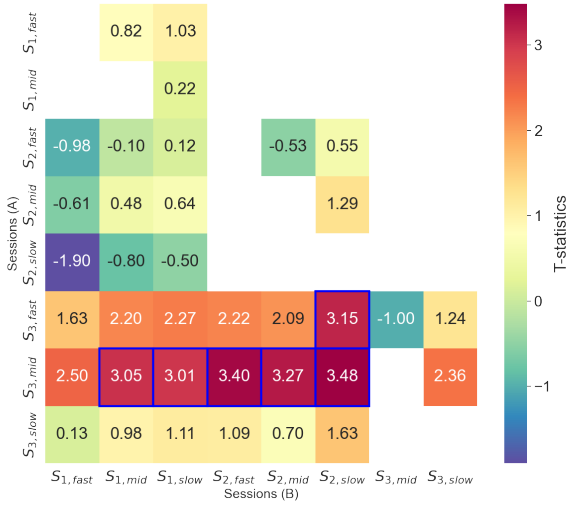


Figure 8: "I felt bored" variances between different test conditions. The Y-axis represents the A group that is tested against B on the x-axis, and the intensity of the color in each cell represents the T-statistics from the test. Cells with blue frames indicate statistical significance ($p < 0.05$).

11. REFERENCES

[1] J. Barbosa, M. M. Wanderley, and S. Huot. Exploring Playfulness in NIME Design: The Case of Live Looping Tools. June 2017.

[2] M. A. Boden and E. A. Edmonds. What is generative art? *Digital Creativity*, 20(1-2):21–46, 2009. Routledge.

[3] L. Callender, C. Hawthorne, and J. Engel. Expanded Groove MIDI Dataset, Apr. 2020.

[4] M. Clayton. What is Entrainment? Definition and applications in musical research. *Empirical Musicology Review*, 7(1-2):49–56, Jan. 2012.

[5] N. M. Collins. *Towards Autonomous Agents for Live Computer Music: Realtime Machine Listening and Interactive Music Systems*. PhD thesis, University of Cambridge, 2006.

[6] J. Côté, J. Baker, and B. Abernethy. Practice and play in the development of sport expertise. In *Handbook of sport psychology, 3rd ed*, pages 184–202. John Wiley & Sons, Inc., Hoboken, NJ, US, 2007.

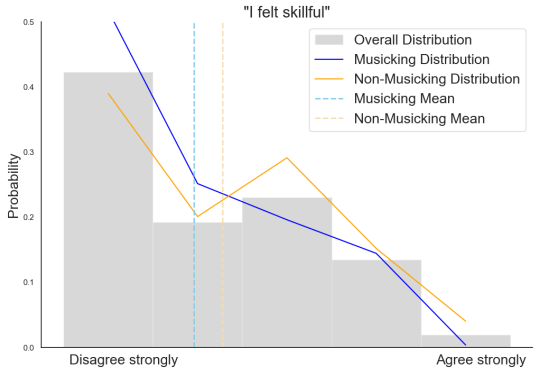
[7] Den Nasjonale forskningsetiske komité for naturvitenskap og teknologi. *Forskningsetiske retningslinjer for naturvitenskap og teknologi*. Oslo, 2nd edition, 2016.

[8] M. Dimock. Defining generations: Where Millennials end and Generation Z begins, 2019.

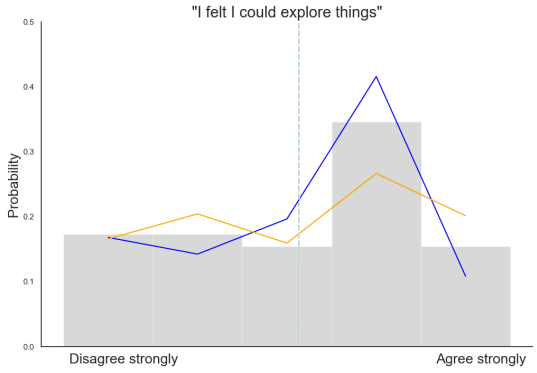
[9] L. Döbereiner. Artistic Potentials of Fallacies in AI Research. Sept. 2022.

[10] C. Erdem. *Controlling or Being Controlled? Exploring Embodiment, Agency and Artificial Intelligence in Interactive Music Performance*. Doctoral thesis, University of Oslo, 2022.

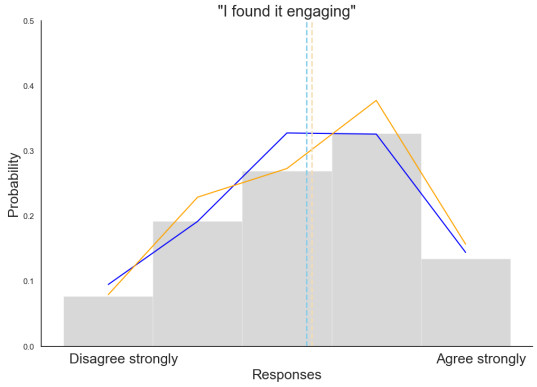
[11] C. Erdem, Q. Lan, J. Fuhrer, C. P. Martin, J. Tørresen, and A. R. Jensenius. Towards Playing in the 'Air': Modeling Motion-Sound Energy Relationships in Electric Guitar Performance Using Deep Neural Networks. In *Proceedings of SMC Conferences*, 2020.



(a)



(b)



(c)

Figure 9: Histograms of distributions of post-study responses on perceived personal skills, explorativeness, and engagement ratings with kernel density plots overlaid for both the Musicking and Non-musicking groups.

[12] C. Erdem, B. Wallace, K. Glette, and A. R. Jensenius. Tool or Actor? Expert Improvisers' Evaluation of a Musical AI "Toddler". *Computer Music Journal*, pages 1–17, Dec. 2023.

[13] C. Erdem, B. Wallace, and A. R. Jensenius. CAVI: A Coadaptive Audiovisual Instrument–Composition. PubPub, June 2022.

[14] E. Frid and A. Ilsar. Reimagining (Accessible) Digital Musical Instruments: A Survey on Electronic Music-Making Tools. In *International Conference on New Interfaces for Musical Expression*, June 2021.

[15] A. Förster, C. Komesker, and N. Schnell. SnoeSky and SonicDive - Design and Evaluation of Two Accessible Digital Musical Instruments for a SEN School. June 2020. Pages: 83–88 Publication Title: Proceedings of the International Conference on New Interfaces for Musical Expression Publisher: Zenodo.

[16] J. Gibbs, Raymond W. *Embodiment and Cognitive Science*. Cambridge University Press, Cambridge, 2005.

[17] J. Gillick, A. Roberts, J. Engel, D. Eck, and D. Bamman. Learning to Groove with Inverse Sequence Transformations, July 2019. arXiv:1905.06118 [cs, eess, stat].

[18] B. Haki, B. Kotowski, C. L. I. Lee, and S. Jordà. TapTamDrum: a dataset for dualized drum patterns. Oct. 2023. Accepted: 2023-10-24T12:25:57Z.

[19] B. Haki, M. Nieto, T. Pelinski, and S. Jordà. Real-Time Drum Accompaniment Using Transformer Architecture. Sept. 2022. Publication Title: Proceedings of the 3rd Conference on AI Music Creativity Publisher: AIMC.

[20] T. Hastie, R. Tibshirani, and J. Friedman. Model Assessment and Selection. In T. Hastie, R. Tibshirani, and J. Friedman, editors, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer Series in Statistics, pages 219–259. Springer, New York, NY, 2009.

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition, Dec. 2015. arXiv:1512.03385 [cs].

[22] A. R. Jensenius. *Sound Actions: Conceptualizing Musical Instruments*. Dec. 2022.

[23] S. Ji, J. Luo, and X. Yang. A Comprehensive Survey on Deep Music Generation: Multi-level Representations, Algorithms, Evaluations, and Future Directions, Nov. 2020. arXiv:2011.06801 [cs, eess].

[24] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, Dec. 2014. arXiv: 1412.6980.

[25] M. Krzyzaniak, C. Erdem, and K. Glette. What Makes Interactive Art Engaging? *Frontiers in Computer Science*, 4, 2022.

[26] E. W. Large. On synchronizing movements to music. *Human Movement Science*, 19(4):527–566, Oct. 2000.

[27] M. Lee, A. Freed, and D. Wessel. Real-Time Neural Network Processing of Gestural and Acoustic Signals. pages 277–280, Montreal, Quebec, Canada, 1991. International Computer Music Association.

[28] M. Lugones. Playfulness, "World"-Traveling, and Loving Perception. In K. Maitra and J. McWeeny, editors, *Feminist Philosophy of Mind*, pages 105–C5.P67. Oxford University PressNew York, 1 edition, July 2022.

[29] D. Makris, M. Kaliakatsos-Papakostas, I. Karydis, and K. L. Kermanidis. Conditional neural sequence

learners for generating drums' rhythms. *Neural Computing and Applications*, 31(6):1793–1804, June 2019.

[30] J. McCormack, T. Gifford, P. Hutchings, M. T. L. Rodriguez, M. Yee-King, and M. d'Inverno. In a Silent Way: Communication Between AI and Improvising Musicians Beyond Sound, Feb. 2019. arXiv:1902.06442 [cs].

[31] A. McPherson, F. Morreale, and J. Harrison. Musical Instruments for Novices: Comparing NIME, HCI and Crowdfunding Approaches. In S. Holland, T. Mudd, K. Wilkie-McKenna, A. McPherson, and M. M. Wanderley, editors, *New Directions in Music and Human-Computer Interaction*, pages 179–212. Springer International Publishing, Cham, 2019. Series Title: Springer Series on Cultural Computing.

[32] T. Nuttall, B. Haki, and S. Jorda. Transformer Neural Networks for Automated Rhythm Generation. In *International Conference on New Interfaces for Musical Expression*, June 2021.

[33] B. H. Repp and Y.-H. Su. Sensorimotor synchronization: A review of recent research (2006–2012). *Psychonomic Bulletin & Review*, 20(3):403–452, June 2013.

[34] A. Schiavio, M. A. G. Witek, and J. Stupacher. Meaning-making and creativity in musical entrainment. *Frontiers in Psychology*, 14, 2024.

[35] O. Senn, L. Kilchenmann, T. Bechtold, and F. Hoesl. Groove in drum patterns as a function of both rhythmic properties and listeners' attitudes. *PLOS ONE*, 13(6):e0199604, June 2018. Publisher: Public Library of Science.

[36] I. Simon, A. Roberts, C. Raffel, J. Engel, C. Hawthorne, and D. Eck. Learning a Latent Space of Multitrack Measures, June 2018. arXiv:1806.00195.

[37] C. Small. *Musicking: the meanings of performing and listening*. Wesleyan University Press, Middletown, Conn., 1998. OCLC: 466918086.

[38] J. Stupacher, T. E. Matthews, V. Pando-Naude, O. Foster Vander Elst, and P. Vuust. The sweet spot between predictability and surprise: musical groove in brain, body, and social interactions. *Frontiers in Psychology*, 13, 2022.

[39] A. Tanaka and M. Donnarumma. The Body as Musical Instrument. *The Oxford Handbook of Music and the Body*, July 2018.

[40] K. Tatar and P. Pasquier. Musical agents: A typology and state of the art towards Musical Metacreation. *Journal of New Music Research*, 48(1):56–105, 2019.

[41] A. Tidemann, P. Öztürk, and Y. Demiris. A Groovy Virtual Drumming Agent. In Z. Ruttkay, M. Kipp, A. Nijholt, and H. H. Vilhjálmsson, editors, *Intelligent Virtual Agents*, Lecture Notes in Computer Science, pages 104–117, Berlin, Heidelberg, 2009. Springer.

[42] E. Tomás. A Playful Approach to Teaching NIME: Pedagogical Methods from a Practice-Based Perspective. 2020.

[43] R. Vogl, H. Eghbal-Zadeh, and P. Knees. An automatic drum machine with touch UI based on a generative neural network. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*, IUI '19, pages 91–92, New York, NY, USA, Mar. 2019. Association for Computing Machinery.

[44] D. Wessel and M. Wright. Problems and Prospects for Intimate Musical Control of Computers. *Computer Music Journal*, 26(3):11–22, 2002. Place: 238 Main St., Suite 500, Cambridge, MA 02142-1046, USA Publisher: MIT Press.

[45] I. Winkler, G. P. Háden, O. Ladinig, I. Sziller, and H. Honing. Newborn infants detect the beat in music. *Proceedings of the National Academy of Sciences*, 106(7):2468–2471, Feb. 2009. Publisher: Proceedings of the National Academy of Sciences.

[46] M. A. G. Witek, E. F. Clarke, M. Wallentin, M. L. Kringelbach, and P. Vuust. Syncopation, Body-Movement and Pleasure in Groove Music. *PLOS ONE*, 9(4):e94446, Apr. 2014. Publisher: Public Library of Science.

[47] M. Wright and E. Berdahl. Towards Machine Learning of Expressive Microtiming in Brazilian Drumming. 2006.

[48] Z. Yin, F. Reuben, S. Stepney, and T. Collins. Deep learning's shallow gains: a comparative evaluation of algorithms for automatic music generation. *Machine Learning*, Mar. 2023.

APPENDIX

A. TEST CONDITIONS

We categorized each test parameter into three distinct levels: low, mid, and high, forming a series of predetermined pairings that contrasted temperature and threshold, with one exception where both parameters were equally set to mid. Table 3 elucidates the interplay among three dimensions: tempo, temperature, and threshold.

The session number (s), temperature (t), and threshold (θ) correlate through the function $C(s, t, \theta)$, which is defined as:

$$C(s,t,\theta) = \begin{cases} \text{slow}, & \text{if } s \in \{1,4,7\}, \\ \text{mid}, & \text{if } s \in \{2,5,8\}, \\ \text{fast}, & \text{if } s \in \{3,6,9\}. \end{cases}$$

Here, a "slow" tempo corresponds to 60 beats per minute (BPM), "mid" to 90 BPM, and "fast" to 120 BPM.

Table 3: Cross-tabulation of session labels with associated temperature and threshold parameters. The table is divided into three temperature categories: low, mid, and high. Each temperature category has three threshold levels: high, mid, and low. The session labels ($S_{x,y}$) denote specific combinations of tempo (slow, mid, fast) and sequence number (1 to 3), arranged vertically by tempo and horizontally by temperature level.

		temperature		
		low	mid	high
		threshold		
		high	mid	low
tempo	slow	$S_{1,slow}$	$S_{2,slow}$	$S_{3,slow}$
	mid	$S_{1,mid}$	$S_{2,mid}$	$S_{3,mid}$
	fast	$S_{1,fast}$	$S_{2,fast}$	$S_{3,fast}$