ohmate as a personal music assistant
What to learn from a creator's choices

Haoran Hou

Not peer reviewed.

Abstract

I want a music system that can learn how I work without taking over the work. This paper develops that aim through ohmate, a proposed assistant for anyone making music. It could draw on choices, edits, references, and optional explanations, alongside movement, humming, and the current project. I distinguish observed events, contextual hypotheses, and the creator's own accounts, then specify a policy for updating and using those records. A worked numerical example follows repeated choices, transfer between projects, unexplained undo, and correction. Under the stated rules, an inferred general preference can be withdrawn while keeping the local choices behind it. The assistant recommends a B version and keeps an optional alternative available; only the creator applies changes to A. In the proposed interaction agreement, dialogue during work serves the current musical task, while questions asked to improve the user model wait until task completion and may be skipped. The evaluation compares recommendations with and without personal history, then tests whether explicit hypotheses help creators find and correct errors. Education and accessibility raise separate requirements for that interaction. This is a conceptual proposal with a reproducible analytical example; the integrated assistant has not been evaluated with users.

Keywords: musical co-creation; personalisation; multimodal interaction; mixed initiative; accessibility

1. Introduction

In my screen-scoring work, I was once asked to complete a large episodic commission under a very tight deadline. That experience made me think seriously about the musical assistant I needed: something that could remember how I work and help me develop ideas, while leaving the decisions to me.

I want ohmate to become a trusted mate for anyone making music: a learner, a performer exploring a sound, or a composer developing an arrangement. What helps will differ between people and tasks.

I imagine entering the studio and finding that the system remembers my harmonic choices, fill revisions, and references. I want to begin with a hummed phrase or movement, without having to translate every idea into technical instructions first.

In an informal camera-to-MIDI test, a movement I intended as vibrato also changed expression and did not reliably follow my hand. The system needed to distinguish what I was using the movement for. Choosing a chord raises a similar question: am I expressing a preference, keeping a passage consistent, or following a brief?

The working project is A. Within an agreed scope, ohmate may prepare B without a detailed request every time. The creator can audition it, keep selected changes, or ask for

© 2026 Haoran Hou.

another direction. They decide how actively the assistant offers help and which changes enter A.

Learning from what happens next is central. Asking about every edit adds work; learning nothing without an explanation misses much of it. How can the assistant use an interpretation while remaining open to correction?

This paper specifies a policy for learning from choices across projects, follows its behaviour through a worked example, and proposes an evaluation of recommendation quality and correction. Explicit hypotheses need not predict better. Their proposed value is that a creator could see which choices support an interpretation, understand where it applies, and revise that scope without losing the original history.

2. Literature Review

2.1. Musical Dialogue, Alternatives, and References

Cococo supports steering and auditioning generated alternatives (Louie et al., 2020). COSMIC uses dialogue state to gather conditions for generation and revision (Zhang et al., 2021). The DAWZY preprint connects text, speech, and humming to REAPER, with project queries and reversible operations (Elkins et al., 2025). These systems support shaping and discussing material within a musical workflow.

Bown et al. (2020) is a particularly close precedent. Their speculative account considers adaptation through user actions, multiple alternatives, and reference music. An observation of professional collaboration shows references being interpreted for selected qualities within a shared task. Building on this work, I focus on revising interpretations of choices across projects and deciding when to ask the creator about them.

2.2. Implicit Evidence and Changing Preferences

MusicRL learns from human comparisons of generated music, alongside rewards for text adherence and audio quality (Cideron et al., 2024). Herremans and Roy (2026) discuss the broader challenge of aligning music generation with human preferences. Explicit comparisons supply a different kind of evidence from an unexplained edit in an ongoing project. The latter can be useful without revealing its cause.

Outside music, Hu et al. (2008) distinguish preference estimates from confidence when learning from implicit feedback. Joachims et al. (2005) show why clickthrough evidence must be interpreted in relation to how alternatives are presented. Coactive learning formalises learning from a user's improvement to a proposal in a particular context, rather than requiring an optimal demonstration or numerical rating (Shivaswamy and Joachims, 2015). These precedents support learning from use. An edit may nevertheless serve a different goal from the one the system assumes, and the edit alone does not reveal which.

Temporal recommendation distinguishes longer-term patterns from transient effects (Koren, 2009). Similar musical choices might follow one project's brief, making time and context relevant to their interpretation. ohmate would retain competing interpretations when they lead to different recommendations, compared with a predictor that uses the same context without explicit hypotheses.

2.3. Initiative and Expressive Interaction

Mixed-initiative interaction treats attention, uncertainty, and the costs of acting or waiting as central concerns (Horvitz, 1999). Guidelines for human-AI interaction include timing, dismissal, correction, and granular feedback (Amershi et al., 2019); COFI places initiative and communication within a broader account of co-creation (Rezwana and Maher, 2022). Notification research in programming and diagram editing shows how task structure and relevance affect interruption management (Iqbal and Bailey, 2008). It motivates attention to interruption costs. For ohmate, I propose an after-task boundary for questions that serve the user model, with the costs discussed in Section 8.

Here, the distinction is between a question that helps finish the current musical task and one asked to improve the assistant's model of the person. The former may be appropriate during work; the latter must wait. This does not prevent learning silently from authorised actions or from an answer volunteered during musical discussion.

Expressive inputs introduce another distinction. Interactive learning and probabilistic movement-to-sound mappings support personal demonstrations and revision through use (Fiebrink et al., 2011; Françoise et al., 2014). A learned mapping still needs a defined role. Oviatt (1999) challenges assumptions that modalities are always combined or redundant: a demonstrated filter gesture, a hummed phrase, and an incidental hand movement each need to be interpreted in relation to the current activity.

2.4. Education and Accessible Creation

Research with novice creators separates obtaining material from knowing how to use it. Fu et al. (2025) examine novice co-creation through retrospective interviews with nine course participants, supported by process records and artefacts. MusicScaffold studies explanations and iterative reflection over four weeks of adolescent music creation (Hu et al., 2026). Roberts et al. (2026) offers a preprint account of one learner's creative process in workshops with tinkerable generative AI. Together, these studies motivate attention to how learners compare, revise, and explain material, beyond whether they accept it.

Accessible music research likewise resists a single ideal interaction. Frid (2019) reviews adaptable interfaces and user involvement. Sound Control explores demonstration-trained mappings, with a preliminary evaluation involving two practitioners adapting controls to children and activities (Parke-Wolfe et al., 2019). The Slide-A-Phone provides a musician-led account grounded in an existing performance practice, including setup and fatigue (McMillan and Morreale, 2025). For ohmate, accessibility must include inspecting, refusing, and correcting assistance, not only providing input. Educational and access needs also complicate inference: a learner's accepted example need not show understanding, and a delayed response need not show disinterest.

3. Learning from Musical Work

3.1. Scope and the Working Project

The initial study would use short passages and a limited library of harmonic, rhythmic, and textural edits. For each proposal, it could then record what was offered, heard, kept, and reversed before extending to less constrained musical work.

The creator chooses what the system may inspect and change in B; protected material remains outside that scope. Each proposal links to its source version of A. If A changes, compatibility must be checked before application. Stopping assistance cancels pending offers without modifying A. Applying a change requires a user action through their chosen accessible control.

Input roles are similarly explicit. Humming may provide rhythmic or melodic material; a calibrated movement may control an effect when enabled. Neither automatically constitutes permission to edit the project. Personalisation should help interpret an authorised input, not make every observable action a command.

3.2. Events, Hypotheses, and Accounts

An *event* records an offer, audition, application, edit, or reversal, with its task, time, project version, and context. A *hypothesis* interprets events, for example as a preference for compact fills when a lead part continues through a transition. A *user account* records what the creator said and its scope. These are not interchangeable accepted/rejected labels.

In the supplement, each hypothesis record stores its contrast, scope, candidate, support score, and supporting event IDs. Opposing choices remain in the event ledger; scope corrections and unresolved disputes are stored separately. Alternative explanations and a full revision history belong to the broader design, not the implemented record. Repeated choices may reflect taste, references, instrument limits, or the task. *Not in this project* is not *never*.

Reference material has a declared or provisional role: drum sound, harmony, pacing, or another quality. A reference for drum sound does not supply fill-length preference votes. Unspecified roles can remain tentative or be clarified to serve the current task. Permission to inspect a reference does not imply permission to use it for unrelated training.

Table 1 summarises what each event contributes to the record. The update rules below specify when that evidence can affect a recommendation.

3.3. A Minimal Update and Recommendation Policy

The minimal policy handles one predefined contrast: compact versus fuller fills. It records possible interpretations at three levels: the current project, an observed musical condition, and a tentative general tendency. Ordinary choices and edits contribute evidence without a questionnaire.

Count choices without multiplying their context. An applied comparison between auditioned options, or the creator's own edit across a defined contrast, supplies a signed choice event. Events sharing a project form one evidence block; projects derived from the same brief, reference, or repeated material are grouped when that provenance is known. Each block contributes the sign of its summed choices: supporting, conflicting, or tied. Thus ten repeated edits do not become ten votes for a general preference. These blocks are bookkeeping units, not statistically independent samples.

For a hypothesis, let S and C count supporting and conflicting blocks. Its support score is

$$q = \max\left(0, \frac{S - C}{S + C + 1}\right). \tag{1}$$

Table 1: Evidence to retain and inferences to avoid. Every event keeps its task and project context.

Event	**Retain**	**Do not infer**
Choice after comparison, or own contrast edit	Auditioned options or edit direction; signed choice in context.	That this choice establishes a lasting preference or its cause.
Single-option application	What was tried and applied.	Preference over an unheard alternative; an extra choice vote.
Unexplained undo	Application and reversal; the implicated interpretation is contested.	Dislike of the music or a known reason for undo.
Unheard dismissal or skipped review	What was dismissed or left unanswered.	A musical-choice vote, confirmation, or resolved uncertainty.
Scoped explanation or correction	The creator's account, affected events, and stated scope.	Permission to extend the account beyond that scope.

The score summarises support, not a probability of knowing intention. Tied blocks contribute to neither count, and hypotheses with $q = 0$ remain dormant. One supporting block permits project-local inference; conditional or general transfer needs at least two from different projects with no known shared provenance.

Select context before using history. Use the most specific eligible, matching record: project, then musical condition, then general. Both directions use the same signed ledger, so at most one direction has positive q at each level; no within-level tie-break is needed here. Each interpretation has a stable key comprising its contrast, scope, scope value, and choice direction. Alternative explanations do not multiply the adjustment.

Limit how much history changes the ranking. First remove candidates violating explicit task constraints. Let $r(c, x)$ be a shared task-relevance score for candidate c in current context x. The example fixes this score to follow the effect of history separately; r excludes the personal-history adjustment calculated below. For one contrast,

$$F(c, x) = r(c, x) + \Delta(c, x), \qquad \Delta(c, x) = \lambda a_h(c) u(h, x) q_h, \tag{2}$$

where h is the selected record and $a_h(c)$ is one for its hypothesised choice and zero for the alternative. Use $\Delta = 0$ if no record applies. For this single contrast, $0 \leq \Delta \leq \lambda$. Here $\lambda = 0.10$ on a unit task-score scale. Constants are design choices for this analysis, not fitted values or validated psychological quantities.

Recommend the highest-scoring admissible candidate as B. Break a tie by the higher task score r, then by ascending candidate ID. The two options differ in fill length. Offer the other for optional audition only if it is admissible and that dimension is allowed; otherwise offer no alternative. With no admissible candidate, offer neither. Neither version changes A automatically.

Treat ambiguity differently from contradiction. Save the interpretation supporting an applied proposal. Unexplained undo preserves the application and choice evidence, but contests that saved interpretation in its task, not whichever record later ranks highest. Set $u(h, x) = 1/2$ only for that task and key; otherwise $u(h, x) = 1$. Repeated undo does not compound attenuation. A genuinely new qualifying choice in the same task, project, lead-presence context, and contrast resolves the dispute in either direction. Duplicate events, single-option trials, unheard dismissals, and skipped questions do not resolve it or supply new choice evidence.

Make correction selective and persistent. Save the account as a restriction on source event IDs, retaining the events. Here it restricts those events to lead-present contexts and removes their general interpretation. Resolving the named task/interpretation dispute requires one account covering its supporting sources: same-direction choices in net-supporting blocks. Other choices remain in the scoring ledger but need not be covered. Reconstruction retains restrictions and unresolved disputes. An integrated assistant would also invalidate dependent B versions and propagate corrections to predictors or caches, leaving A unchanged. Current instructions take precedence.

3.4. Musical Dialogue During Work, Learning Questions Afterward

Asking whether a fill should leave room for the lead part may resolve the current arrangement. Asking about a lasting taste after that passage is settled serves the assistant's learning. Under the proposed agreement, the first question is allowed, subject to initiative and quiet settings; the second waits until task completion. Background learning from authorised edits or volunteered explanations can continue. Unanswered task questions require a provisional B or a pause in that proposal, rather than repeated prompting.

The creator explicitly marks task completion; a save, playback stop, or idle period is insufficient. Review is optional and never a condition of saving, exporting, or further help. Skipping it supplies no confirmation or repeated request; starting another task defers remaining questions. During work, questions must address a concrete unresolved musical choice, without an appended profile questionnaire.

Initiative settings govern requested, quietly prepared, or proactively offered help. Quiet mode stops unsolicited dialogue. Disabling review leaves background learning and requested help available, and creators may initiate correction at any time.

4. A Worked Analysis: What Changes after a Correction

Consider a constructed example across two music projects. A creator shortens fills in a vocal-band demo and an electronic-pop song with separate briefs and references. Both have lead material across the transition. Compact fills may suit that situation, or the creator may generally prefer them. After the second evidence block, the policy permits the latter interpretation to influence a new project while retaining both readings.

Two admissible candidates use the same drum sound: compact (K) and fuller (F). Set their task-relevance scores to $r(K) = 0.56$ and $r(F) = 0.60$ to follow the arithmetic. These are illustrative inputs, not measured musical quality; a request for energy need not favour a longer fill.

Table 2: Constructed policy trace, not participant results. K is compact and F fuller. The primary B is listed first; the other remains optionally auditionable. Scores are rounded to three decimals.

State / query	**Applicable support**	**Scores K/F**	**B / alt.**
One source project; lead continues	Local, $q = 1/2$	0.610/0.600	K/F
Two source projects; new-project instrumental task	General, $q = 2/3$	0.627/0.600	K/F
Same task, after unexplained undo	General, $uq = 1/3$	0.593/0.600	F/K
After scoped account; same instrumental task	No matching record	0.560/0.600	F/K
After same account; different project, lead present	Conditional, $q = 2/3$	0.627/0.600	K/F

Table 2 uses these same task scores throughout, two positive source blocks with no conflicts, and no other records. With one block, $q = 1/2$; with two, $q = 2/3$. For an instrumental task in a new project, only the tentative general scope matches. The creator tries K alone and applies it, then silently undoes it. Without comparing options or making their own contrast edit, this application creates neither a third preference-evidence block nor a local preference record. It remains logged; undo contests the implicated interpretation without establishing dislike.

After finishing the task, the creator optionally explains: *Those short fills left space for the lead parts. They are not my general preference.* The instrumental query loses its adjustment; a different lead-present project retains it. Reconstruction from the saved events, correction ledger, and disputes preserves scores when query, task scores, constraints, and parameters are unchanged. This reproduces both the pre-account unresolved-undo state (0.593/0.600) and the corrected queries (0.560/0.600 and 0.627/0.600). The corrected events cannot recreate the unrestricted general record. Skipping review leaves the dispute unresolved.

Scope correction takes effect before dispute resolution. Restricting only e_1 leaves one general-support block, below the two-block threshold: K falls to 0.560, while the dispute remains unresolved because e_2 is not covered. F remains first. Reconstruction preserves both this score and the pending dispute.

A no-history baseline sets $\Delta = 0$ throughout, with the same scores, candidates, and constraints. It ranks F first at 0.560/0.600, isolating history's effect on ranking, not musical usefulness.

4.1. What the Example Shows

A direct contextual policy could reproduce these scores through scoped choice counts and undo attenuation, without competing reason records. The supplement supplies a stipulated score table as an equivalence witness, not a trained predictor. Observed-choice learning could retain contextual counts and honour correction without reproducing the undo switch. These two learners remain proposed study conditions; the implemented policies are the minimal revisable-hypothesis policy and the no-history baseline.

The record identifies which events to restrict and which queries to check. Direct predictors must also be allowed comparable attribution and correction. The instrumental query tests removal of the unwanted generalisation, the lead-present query tests retention of relevant evidence, and reconstruction tests that the same events cannot restore the rejected interpretation.

The central result concerns the structure of the record. Retaining local choices and preventing reconstruction from restoring a rejected generalisation follow from storing events separately from interpretations and enforcing corrections when rebuilding records. These properties do not depend on the illustrative values $\lambda = 0.10$ or $u = 1/2$. The numerical ranking does. Compact becomes primary only when $\lambda uq > 0.04$; at equality the higher task score favours fuller. With $q = 2/3$, that requires $\lambda > 0.06$ before undo and $\lambda > 0.12$ afterward. The chosen 0.10 produces the switch, while other parameters or task scores may not. Repeating a supporting choice within an already supporting block adds no evidence block. Newly identified shared provenance can merge blocks and withdraw transfer eligibility.

The Python supplement reproduces all five states and tests interpretation-specific undo, unresolved-state reconstruction, selective resolution, provenance grouping, allowed alternatives, constraints, and both candidate-ranking tie-breaks. It uses exact fractions for scoring and comparisons; decimals appear only in the displayed results.

5. Education: Help without Doing the Learning

In education, the task may be hearing a difference rather than finishing a track quickly. ohmate might offer two harmonisations, let a learner compare them, and answer questions about their effects. Work on novice co-creation and scaffolding motivates attention to revision and reflection, not a presumption that generated examples teach by themselves (Fu et al., 2025; Hu et al., 2026; Roberts et al., 2026).

A question that helps a student investigate a musical problem belongs to that activity and may occur during it. Under the proposed agreement, a question asked only to improve the assistant's profile waits until the activity ends.

A learner may accept an example to copy it, experiment with it, or because they lack a way to disagree. To assess understanding, an educational study would examine explanations, independent revision, and a later task without musical suggestions, while retaining required access controls.

6. Accessibility: Control over the Whole Interaction

An assistant for anyone making music cannot depend on one ideal gesture, voice, response speed, or vocabulary. Accessible music research motivates adaptable controls and involve-

ment of the people concerned (Frid, 2019; Parke-Wolfe et al., 2019; McMillan and Morreale, 2025). This includes requesting, auditioning, refusing, partially applying, undoing, and correcting assistance.

Alternative inputs are not automatically accessible. A gesture may be tiring, speech unavailable, or comparison controls difficult to navigate. Delay may reflect interface cost rather than musical uncertainty. It should not become evidence of dislike; declining review must not reduce access to assistance.

If only verbal explanations can change a hypothesis, the design gives some creators less control. Options such as *only this project*, *not my preference*, and *forget this interpretation* could support non-verbal correction and should be developed with the creators who would use them. The controls must also distinguish operating the DAW from authorising a compositional choice.

7. Proposed Evaluation

The proposed studies ask whether personal history improves musical suggestions, and whether an explicit record of interpretations makes mistakes easier to find and correct. Recommendation quality and correction effort are measured separately.

7.1. Policies and Shared Information

Four conditions would share current requests, reference roles, candidate library, task features, explicit task constraints, and interface controls. All offer optional alternatives, require human application, and respect the same interaction boundaries.

No-history ranking would use the current task score $r(c, x)$ with $\Delta = 0$, excluding stored personal choices and inferred preferences. It still honours instructions and corrections supplied for the current task. This condition tests whether personal history adds value beyond understanding the present request.

The other three conditions would receive the same authorised personal history and accounts. *Observed-choice learning* would use selected edits and accounts in context, leaving unexplained undo and dismissal without a reason label. *Direct contextual prediction* would also use ambiguous behaviour with confidence and context, without explicit competing reason records. *Revisable hypotheses* would use the mechanism in Section 3.3, with its evidence and scope records available for correction. Each personalised condition must support task-dependent preferences and scope-aware correction.

Personal history would enter each ranking once, through its preference adjustment; the shared base score remains non-personalised. Before evaluation, each implementation, grouping rule, adjustment bound, and tuning budget would be fixed. Model capacity, inference time, and correction costs must be reported alongside the outcomes.

7.2. Recommendation and Selective Correction

A common collection interface would record proposals, auditions, choices, and edits, with optional accounts after explicit task completion. Earlier tasks would train a policy and later tasks test it. Related passages and projects sharing a brief or source material stay within the same split. A later account cannot enter an earlier prediction.

Counterbalanced, matched tasks would continue a direction, follow a new reference, or depart from a past pattern. All four conditions enter this recommendation comparison. Creators would request alternatives at chosen moments, holding offer timing constant. Recording the current direction before audition helps distinguish a faithful suggestion from a useful surprise that changes the goal. Relevance and editing effort would be assessed alongside adoption.

A recovery test would supply the same misleading history and later scoped correction to the three personalised policies. It would examine removal of an unwanted tendency in the corrected context, preservation of useful behaviour elsewhere, and replay without restoring the rejected interpretation. The no-history condition has no learned interpretation to repair and is excluded from this comparison. Injecting a wrong hypothesis directly would be a separate component stress test; the shared-history test assesses errors arising from each policy's own use of the evidence.

The inspection study would use matched initial recommendations and the same history view, allowing each personalised policy to expose its supporting information. Participants would identify the source and scope of an error, correct it, and check both contexts. Measures include actions, time, unintended loss of useful preferences, and verification of the repair. An ablation would keep the same backend but hide the editable reason record, separating its interaction value from ranking changes.

A further ablation would withhold retrospective accounts from the personalised conditions while retaining explicit current-task instructions. It would assess usefulness, supervision burden, and recovery from later evidence without scoring a system against an unobserved motive.

7.3. Initiative, Analysis, and Limits

A separate study would hold ranking constant and compare requested-only help, fixed offer rules, and contextual offer timing. All preserve the creator's adjustable initiative, quiet control, and after-task boundary for learning questions. Logs distinguish preparation, offer, audition, and application. Optional post-task judgments address content and timing separately. Outcomes include unwelcome interruption, wanted help that never arrived, and review effort: fewer offers alone rewards a silent assistant.

Missing retrospective answers are expected, not failed participation. Analysis would report review use and whether usefulness depends on it. Zero learning-only prompts before explicit task completion measures compliance, not optimal timing. Task-serving questions also need checking for a concrete unresolved musical choice.

A pilot would establish tasks and measures before fixing the main sample size. Primary comparisons, exclusions, and update rules would be declared in advance. Proposals are repeated observations within creators, not independent participants; estimates need uncertainty. Offline replay cannot establish reactions to unheard alternatives. Ethics review or an institutional determination would precede recruitment, with control over recordings and retained history. Education and access-specific co-design require separate protocols.

8. Discussion

ohmate should use earlier choices without treating them as a fixed description of the person. The example withdraws a general preference while retaining its contextual use. A direct contextual model can produce the same rankings; the explicit record must therefore be tested for its value in finding and correcting errors.

An explanation can sound convincing and still be wrong. Creators should see its source events, reject it without justification, and have that correction affect later recommendations. They should also be able to use B and its alternative without managing this history.

Separate projects may share unknown constraints, so the evidence-block rule cannot remove all confounding. Newly identified shared provenance should merge blocks and trigger recomputation. Recurring keys in sad pieces, for example, might reflect vocal range rather than an emotional preference. Contextual association alone cannot settle that question.

Waiting until the task ends has costs: people may forget what they meant, skip review, or find another completion control inconvenient. I chose this boundary to protect work in progress. It should be evaluated on those terms, and the assistant must remain useful when no retrospective explanation is given.

EEG is a separate longer-term question. MoodMixer provides a precedent for collaborative EEG sonification (Leslie and Mullen, 2011); Di Liberto et al. (2021) classify recordings of professional musicians repeatedly hearing or imagining four known melodies. Neither demonstrates unrestricted musical thought decoding. A future study could ask whether EEG adds evidence about responses to an audition beyond audio, context, and feedback, accounting for movement artefacts, held-out sessions, and setup burden. It could not override the creator's account or authorise changes to A.

The numerical example checks rules, not musical quality, effort, or interface experience. Fill length is an illustration; harmony, rhythm, and texture need validated task scores and ways to handle interacting dimensions. Education and accessibility need studies with the people concerned. The next step is to test whether correction controls help people make music and how much work they require.

9. Conclusion

I want ohmate to remember how someone works while leaving room for different choices. The example preserves a correction across contexts and reconstruction without discarding relevant history. Evaluation must test recommendation quality and the value of explicit hypotheses for correcting errors. The assistant prepares alternatives; the creator decides what enters the project. Model-learning questions remain optional and wait until task completion.

Acknowledgments

Generative AI disclosure. ChatGPT assisted with manuscript organisation and language editing, refinement of the proposed rules, checking the worked example, and developing the tables and supplementary code. No participant data or empirical findings were generated.

References

Amershi, S., Weld, D., Vorvoreanu, M., Fourney, A., Nushi, B., Collisson, P., Suh, J., Iqbal, S., Bennett, P. N., Inkpen, K., Teevan, J., Kikin-Gil, R., and Horvitz, E. (2019). Guidelines for Human-AI Interaction. Proceedings of CHI, 1-13. https://doi.org/10.1145/3290605.3300233

Bown, O., Grace, K., Bray, L., and Ventura, D. (2020). A Speculative Exploration of the Role of Dialogue in Human-Computer Co-creation. Proceedings of ICCC, 25-32. https://computationalcreativity.net/iccc20/papers/046-iccc20.pdf

Cideron, G., Girgin, S., Verzetti, M., Vincent, D., Kastelic, M., Borsos, Z., McWilliams, B., Ungureanu, V., Bachem, O., Pietquin, O., Geist, M., Hussenot, L., Zeghidour, N., and Agostinelli, A. (2024). MusicRL: Aligning Music Generation to Human Preferences. Proceedings of ICML, PMLR 235, 8968-8984. https://proceedings.mlr.press/v235/cideron24a.html

Di Liberto, G. M., Marion, G., and Shamma, S. A. (2021). Accurate Decoding of Imagined and Heard Melodies. Frontiers in Neuroscience, 15, 673401. https://doi.org/10.3389/fnins.2021.673401

Elkins, A. C., Singh, S., Kieback, A., Blankenship, S., Amadasun, U. P., and Chadha, A. (2025). DAWZY: A New Addition to AI powered "Human in the Loop" Music Co-creation. arXiv preprint, version 1. https://arxiv.org/abs/2512.03289v1

Fiebrink, R., Cook, P. R., and Trueman, D. (2011). Human Model Evaluation in Interactive Supervised Learning. Proceedings of CHI, 147-156. https://doi.org/10.1145/1978942.1978965

Françoise, J., Schnell, N., Borghesi, R., and Bevilacqua, F. (2014). Probabilistic Models for Designing Motion and Sound Relationships. Proceedings of NIME, 287-292. https://doi.org/10.5281/zenodo.1178764

Frid, E. (2019). Accessible Digital Musical Instruments–A Review of Musical Interfaces in Inclusive Music Practice. Multimodal Technologies and Interaction, 3(3), 57. https://doi.org/10.3390/mti3030057

Fu, Y., Newman, M., Going, L., Feng, Q., and Lee, J. H. (2025). Exploring the Collaborative Co-Creation Process with AI: A Case Study in Novice Music Production. Proceedings of DIS, 1298-1312. https://doi.org/10.1145/3715336.3735829

Herremans, D., and Roy, A. (2026). Aligning Generative Music AI with Human Preferences: Methods and Challenges. Proceedings of the AAAI Conference on Artificial Intelligence, 40(46), 39699-39706. https://doi.org/10.1609/aaai.v40i46.41323

Horvitz, E. (1999). Principles of Mixed-Initiative User Interfaces. Proceedings of CHI, 159-166. https://doi.org/10.1145/302979.303030

Hu, Y., Koren, Y., and Volinsky, C. (2008). Collaborative Filtering for Implicit Feedback Datasets. Proceedings of IEEE ICDM, 263-272. https://doi.org/10.1109/ICDM.2008.22

Hu, Z., Liu, Y., Zhang, Z., Chen, G., Yu, B., Li, J., and Cao, J. (2026). MusicScaffold: Bridging Machine Efficiency and Human Growth in Adolescent Creative Education through Generative AI. Proceedings of CHI, Article 1084, 1-21. https://doi.org/10.1145/3772318.3790440

Iqbal, S. T., and Bailey, B. P. (2008). Effects of Intelligent Notification Management on Users and Their Tasks. Proceedings of CHI, 93-102. https://doi.org/10.1145/1357054.1357070

Joachims, T., Granka, L., Pan, B., Hembrooke, H., and Gay, G. (2005). Accurately Interpreting Clickthrough Data as Implicit Feedback. Proceedings of SIGIR, 154-161. https://doi.org/10.1145/1076034.1076063

Koren, Y. (2009). Collaborative Filtering with Temporal Dynamics. Proceedings of SIGKDD, 447-456. https://doi.org/10.1145/1557019.1557072

Leslie, G., and Mullen, T. (2011). MoodMixer: EEG-based Collaborative Sonification. Proceedings of NIME, 296-299. https://www.nime.org/proceedings/2011/nime2011_296.pdf

Louie, R., Coenen, A., Huang, C. Z., Terry, M., and Cai, C. J. (2020). Cococo: AI-Steering Tools for Music Novices Co-Creating with Generative Models. IUI 2020 Workshops, Workshop on Human-AI Co-Creation with Generative Models. CEUR Workshop Proceedings, 2848, 6 pp. https://ceur-ws.org/Vol-2848/HAI-GEN-Paper-1.pdf

McMillan, A., and Morreale, F. (2025). The Slide-A-Phone: A Tactile Accessible Musical Instrument. Proceedings of NIME, 218-223. https://doi.org/10.5281/zenodo.15698839

Oviatt, S. (1999). Ten Myths of Multimodal Interaction. Communications of the ACM, 42(11), 74-81. https://doi.org/10.1145/319382.319398

Parke-Wolfe, S. T., Scurto, H., and Fiebrink, R. (2019). Sound Control: Supporting Custom Musical Interface Design for Children with Disabilities. Proceedings of NIME, 192-197. https://doi.org/10.5281/zenodo.3672920

Rezwana, J., and Maher, M. L. (2022). Designing Creative AI Partners with COFI: A Framework for Modeling Interaction in Human-AI Co-Creative Systems. arXiv preprint, version 1. https://arxiv.org/abs/2204.07666v1

Roberts, C. L., Fasiang, K., Miller, C., O'Rourke, E., and Horn, M. (2026). Scaffolding the Creative Process in Music with Tinkerable Generative AI. SSRN preprint. https://doi.org/10.2139/ssrn.6756304

Shivaswamy, P., and Joachims, T. (2015). Coactive Learning. Journal of Artificial Intelligence Research, 53, 1-40. https://doi.org/10.1613/jair.4539

Zhang, Y., Xia, G., Levy, M., and Dixon, S. (2021). COSMIC: A Conversational Interface for Human-AI Music Co-Creation. Proceedings of NIME, Article 41. https://doi.org/10.21428/92fbeb44.110a7a32