Music Atelier: Exploring the Knowing–Doing Gap in LLM Creativity via Symbolic Music Composition

Zhejing Hu[1], **Yan Liu**[1*], **Zhi Zhang**[1], **Sean Fontaine**[2] and **Gong Chen**[2]
[1] Department of Computing, The Hong Kong Polytechnic University
[2] Clozzz Lab
zhejing.hu@connect.polyu.hk, yan.liu@polyu.edu.hk, zhi271.zhang@connect.polyu.hk, sfontaine20@ucla.edu, heinz@clozzz.com

Abstract

The knowing–doing gap, the mismatch between ideas articulated during model reasoning and the realized creative artifact, remains a fundamental challenge in creative AI and persists in LLM-based artistic creation. This paper responds to this gap by introducing a systematic and interpretable framework for examining how user intent is articulated during model reasoning and selectively realized, or lost, during action in LLMs, using symbolic music composition as an analytical lens. We present a realizational process theory that formalizes creative generation and localizes the knowing–doing gap at the realization stage. We instantiate this theory with Music Atelier (Mutelier), an LLM-as-a-Judge framework that operationalizes idea-level realization analysis and makes the gap observable and analyzable in practice. Across diverse evaluation settings, we show that Mutelier reveals reliable, previously invisible failure modes in which intent-aligned ideas are articulated during reasoning but fail to materialize in the final artifact. By reframing artistic creation as a realizational process, this work provides a principled process-level foundation for understanding how knowing–doing gaps emerge in machine creativity.

1 Introduction

The maxim "knowing is easy, doing is hard" has long been recognized in philosophical traditions as capturing the enduring gap between understanding and action [Crisp, 2014]. In creative practice, theoretical understanding or technical mastery is often only a starting point, rather than a guarantee of successful artistic realization. Artistic creation therefore centers on this process of transformation, through which understanding is carried into practice and ideas become concrete works.

As large language models (LLMs) become deeply embedded in creative tasks [Peng, 2022; Huang *et al.*, 2025; Wan *et al.*, 2024], the knowing–doing gap takes on a new, model-specific form. In this context, LLMs often make user intent and intent-relevant knowledge explicit through reasoning traces that outline how a creative task could be carried out ("knowing") [Wei *et al.*, 2022], and subsequently produce a final generated artifact as the realization of this reasoning ("doing") [Zhou *et al.*, 2024; OpenAI, 2025b]. However, it remains unclear how user intent, as expressed through intermediate creative ideas during model reasoning, is translated, or fails to be translated, into concrete elements of the generated artifact in a systematic and analyzable way. This leaves a gap in creative research, reflected in the lack of interpretable and tractable representations that allow humans to observe, understand, and meaningfully interact with how a model applies its articulated knowledge during the creative process to shape identifiable elements of an artistic artifact.

When applied to symbolic music composition, the knowing–doing gap becomes particularly salient. LLMs can competently articulate music-theoretic knowledge, such as harmony, tonality, and rhythm, and reason about how these concepts should be applied to construct musical structure [Wang *et al.*, 2025a; OpenAI, 2025a]. However, the explicit articulation of such knowledge does not ensure its consistent realization in the generated music. In practice, reasoning traces that describe coherent musical plans often fail to align with the concrete musical elements instantiated in the final artifact, revealing a structural mismatch between articulated theory and realized composition. Because music composition relies on rule-guided rather than rule-determined creation, symbolic music provides a natural testbed for examining how knowing–doing gaps emerge between planning and realization.

Prior analysis paradigms for music composition primarily focus on evaluating final artifacts, emphasizing output quality or preference alignment [Bian *et al.*, 2023; Hu *et al.*, 2023; Cideron *et al.*, 2024; Hu *et al.*, 2025b]. Attribute-level statistics and LLM-based judging pipelines further provide useful descriptive and post-hoc perspectives on generated music [Wang *et al.*, 2021; Yuan *et al.*, 2023; Zheng *et al.*, 2023]. While effective for outcome-oriented evaluation, these artifact-centered approaches are not explicitly designed to trace how a model's articulated reasoning or knowledge is realized during the creative process, leaving open questions about the correspondence between reasoning and realized decisions. This motivates the need for a complementary process-level analytical framework.

We introduce a process-level perspective on creative gen-

*Corresponding author.

eration that treats artistic creation as a realizational process linking user intent, articulated reasoning, and the final artifact. Building on this perspective, we propose Music Atelier (Mutelier), a systematic analytical framework that makes the knowing–doing gap in creative LLMs explicit by tracing how intent-aligned ideas are articulated during reasoning and selectively realized, or lost, in generated artworks. At a technical level, Mutelier instantiates a novel LLM-as-a-Judge paradigm that treats creative realization itself as the object of analysis rather than merely evaluating final outcomes. We validate Mutelier through a series of controlled studies across three creative tasks and fifteen LLMs, through which Mutelier reveals consistent patterns in how creative intentions fail to materialize in practice—patterns that remain largely invisible to artifact-centered analyses. By treating the knowing–doing gap as a realizational phenomenon, this work provides the field with a principled process-level lens for studying machine creativity beyond artifact-centered evaluation.

2 Related Work

2.1 LLM-as-a-Judge

Recent advances in LLMs have given rise to the LLM-as-a-Judge paradigm, where LLMs are employed as general-purpose evaluators or decision-makers in generative systems [Wang *et al.*, 2023; Panickssery *et al.*, 2024; Chen *et al.*, 2024; Desmond *et al.*, 2025; Zhang *et al.*, 2025; Hu *et al.*, 2025a]. In this line of work, LLMs are primarily used to approximate human judgments over generated outputs, such as preference alignment, factual correctness, or task compliance, enabling scalable evaluation without extensive human annotation [D'Souza *et al.*, 2025; Ye *et al.*, 2025]. Empirical studies show that models such as GPT-4 can achieve strong agreement with human judges in benchmarks like MT-Bench and Chatbot Arena [Zheng *et al.*, 2023], while subsequent frameworks further improve robustness and consistency in judgment output quality [Yu *et al.*, 2025]. Despite their effectiveness, prior LLM-as-a-Judge approaches implicitly assume a closed artifact as the object of judgment, where an output is generated first and then evaluated against a criterion. In contrast, the problem we study requires judging the correspondence between articulated reasoning and its realization in the generated output, making the judgment object a cross-stage mapping rather than a single artifact. Unlike prior LLM-as-a-Judge work that treats the model mainly as a scoring or evaluation tool for final outputs, we extend this paradigm by using LLM judgment as an analytical operator that explicitly analyzes cross-stage mappings, enabling principled study of how the knowing–doing gap manifests in creative AI.

2.2 Creativity in Generative Models

Research on creativity in generative models has predominantly focused on properties of generated outputs, such as novelty, diversity, surprise, or stylistic deviation [Friedman and Dieng, 2023; Zhang *et al.*, 2024; Zhong *et al.*, 2024; Franceschelli and Musolesi, 2024]. Some studies draw on psychological theories of creativity and apply multi-dimensional annotations or ratings to artifacts [Chakrabarty *et al.*, 2024], while others design structured tasks to indirectly probe associative reasoning or conceptual recombination abilities [Huang *et al.*, 2025]. Together, these approaches provide valuable perspectives for characterizing creative performance across domains [Meng *et al.*, 2025; Wang *et al.*, 2025b]. However, most existing work treats creativity as an observable attribute of final outputs, rather than as a dynamic process unfolding during generation. Such outcome-centered perspectives make it difficult to examine how creative ideas emerge, how knowledge is invoked, or how abstract intentions are translated into concrete generative realizations. In contrast, this paper adopts a process-oriented view of creativity, framing creative generation as a realizational phenomenon that links reasoning-stage ideas to their manifestations in artifacts.

3 Mutelier

We develop a process-level realizational theory of creative generation and introduce Mutelier.

3.1 A Realizational Process Theory

We conceptualize creative generation in LLMs as a *realizational process* across three stages (Figure 1): (i) user intent specification, (ii) idea articulation during reasoning, and (iii) artifact realization. Formally, a creative generation instance is represented as a triplet (x, r, y), where x denotes the user input encoding creative intent, r denotes the unstructured reasoning trace produced by the model, and y denotes the final generated artifact. This formulation explicitly separates internal idea articulation from external realization.

Within this framework, we define the *knowing–doing gap* as a realization failure at the reasoning-to-artifact transition. Let $\mathcal{I}(r) = \{i_k\}$ denote the set of creative ideas articulated in a reasoning trace r, and let $\rho(i_k, y) \in \{0, 1\}$ indicate whether an articulated idea i_k is realized in the artifact y. The knowing–doing gap occurs if there exists a non-empty subset $\mathcal{I}^+ \subseteq \mathcal{I}(r)$ such that each $i_k \in \mathcal{I}^+$ is aligned with the user intent x but $\rho(i_k, y) = 0$. This localizes the gap to the $r \to y$ transition and necessitates idea-level analysis.

3.2 The Thought Graph as an Intermediate Representation

To operationalize the realizational process theory defined above, we introduce the *thought graph*, an explicit intermediate representation that instantiates articulated creative ideas as analyzable structural units.

A thought graph is defined as a directed graph

$$G = (N, L), \tag{1}$$

where each node $n_i \in N$ corresponds to a distinct creative idea articulated during reasoning. Formally, the node set N provides a structured representation of the idea set $\mathcal{I}(r)$, with each node n_i corresponding to an element i_k. Each directed edge $(u, v) \in L$ captures a structural relation between ideas (e.g., refinement, dependency, or progression).

Each idea node n_i is annotated with two complementary attributes:

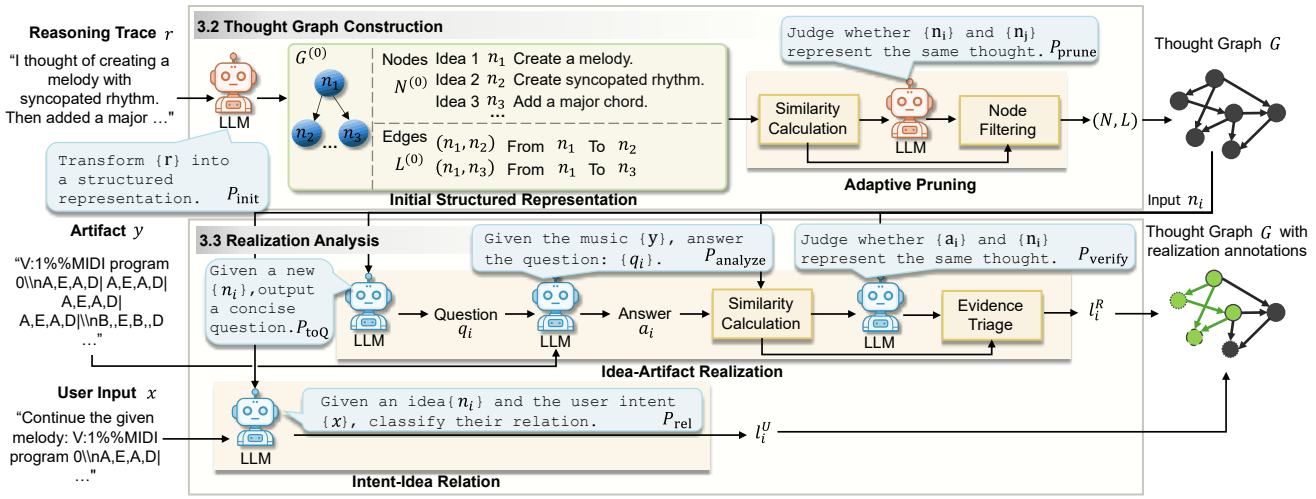


Figure 1: Overview of Mutelier. Creative generation is modeled as a realizational process spanning user intent, intermediate reasoning, and artifact realization. Node colors in the thought graph denote idea–artifact realization labels, while node border shapes represent intent–idea relation labels.

- an *idea–artifact realization label* $\ell_i^R \in \{0, 1\}$, which instantiates the realization indicator $\rho(i_k, y)$ for the corresponding idea, specifying whether it is realized in the final artifact y;
- an *intent–idea relation label* $\ell_i^U \in \mathcal{K}$, which maps each idea node to a relation category in $\mathcal{K}$ with respect to the user intent x (e.g., preserve, elaborate, shift, or diverge).

We next describe how this abstract representation is instantiated from an unstructured reasoning trace r. Given an unstructured reasoning trace r, the goal of graph construction is to extract a structured representation that captures the model's articulated ideas and their relations. We first generate an initial graph

$$G^{(0)} = \text{LLM}(r;\, P_{\text{init}}), \qquad G^{(0)} = (N^{(0)}, L^{(0)}), \tag{2}$$

where the prompting constraint P_{init} induces the LLM to output idea nodes and directed relations. Each node in $N^{(0)}$ is a short textual description of a creative idea, and each edge in $L^{(0)}$ denotes a structural relation between ideas.

Raw reasoning traces often contain redundant or overlapping idea articulations. To obtain a semantically minimal yet expressive representation, we apply an adaptive pruning procedure. For all node pairs $(n_i, n_j) \in N^{(0)}$ with $i < j$, we compute

$$s_{ij}^P = \text{Sim}(n_i, n_j), \tag{3}$$

where $\text{Sim}(\cdot, \cdot)$ denotes cosine similarity between sentence embeddings. An adaptive threshold $\tau_P = \mu + \sigma$ is computed from the distribution of $\{s_{ij}^P\}$. For node pairs satisfying $s_{ij}^P > \tau_P$, we query an LLM with prompt P_{prune} to determine whether the nodes express the same underlying idea. Confirmed duplicates are removed, yielding a pruned graph $G = (N, L)$ that serves as the basis for all subsequent analysis. Table 2 reports the effect of adaptive pruning on redundancy reduction in the thought graph.

3.3 Realization-Centered Dual-Dimension Analysis

With the annotated thought graph, we analyze the realizational process along two complementary dimensions: idea–artifact realization and intent–idea relation. Together, these dimensions operationalize the abstract mappings defined in the realizational process theory.

Idea–Artifact Realization

For each idea node n_i corresponding to an articulated idea $i_k \in \mathcal{I}(r)$, we determine whether it is realized in the artifact y, thereby instantiating the realization indicator $\rho(i_k, y)$. To ensure rigorous and conservative judgment, we adopt a strategy that decouples evidence extraction from realization verification.

Specifically, each idea is first converted into a verifiable, artifact-grounded question

$$q_i = \text{LLM}(n_i;\, P_{\text{toQ}}), \tag{4}$$

which is answered using the artifact as contextual evidence:

$$a_i = \text{LLM}(y, q_i;\, P_{\text{analyze}}). \tag{5}$$

We then compute a semantic consistency score

$$s_i^R = \text{Sim}(n_i, a_i), \tag{6}$$

which serves as a continuous measure of evidential support for realization rather than a direct realization decision. Based on the distribution of $\{s_i^R\}$, we adopt a two-threshold evidence triage strategy with $\tau_{\text{hi}} = \mu + \sigma$ and $\tau_{\text{lo}} = \mu - \sigma$. Only ambiguous cases are passed to explicit verification:

$$v_i = \text{LLM}(n_i, a_i;\, P_{\text{verify}}), \qquad v_i \in \{0, 1\}. \tag{7}$$

The final realization label is defined as

$$\ell_i^R = \begin{cases} 1, & s_i^R > \tau_{\text{hi}}, \\ 1, & \tau_{\text{lo}} \le s_i^R \le \tau_{\text{hi}} \ \wedge \ v_i = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

This procedure yields a conservative instantiation of the realization function ρ at the level of individual ideas. Table 3 empirically validates its effectiveness in reducing false positives.

Intent–Idea Relation

Each idea node $n_i \in N$ is independently classified according to its semantic relation to the original user intent x:

$$\ell_i^U = \mathrm{LLM}(n_i, x;\, P_{\mathrm{rel}}), \tag{9}$$

thereby instantiating the intent–idea alignment relation required for conditioning realization analysis.

Finally, given intent and realization labels on the thought graph, we summarize realization outcomes conditioned on intent–idea relations. For each relation category $c \in \mathcal{K}$, we define

$$\Phi(x, y; G) = \left(|\{n_i \in N \mid \ell_i^U = c \wedge \ell_i^R = 1\}|\right)_{c \in \mathcal{K}}, \tag{10}$$

where $\Phi(x, y; G)$ constitutes a structural characterization of the knowing–doing gap, revealing how different classes of intent-aligned ideas are selectively realized, weakened, or lost during execution, rather than a single scalar performance score.

4 Experiment

Our experiments consist of three parts: (1) validating Mutelier by comparison with human expert annotations; (2) analyzing the knowing–doing gap across current state-of-the-art LLMs; and (3) demonstrating the practical usefulness of Mutelier for human users.

4.1 Experimental Setups

Tasks and Data. We study three representative symbolic composition settings: melody-to-accompaniment generation, melody continuation, and text-to-music composition. For first two tasks, we use phrase-level data from Pop909 [Wang *et al.*, 2020b; Hu *et al.*, 2024]. For the third task, we sample prompts from MidiCaps [Melechovský *et al.*, 2024], each containing at least two controllable attributes (e.g., mood, tempo). Each task includes 150 test samples, balancing diversity with feasibility for expert annotation [Chakrabarty *et al.*, 2024].

Compared Analysis Methods. We compare three analysis configurations under the same backbone (GPT-4o). *Output-oriented* analysis evaluates the (x, y, r) pair without eliciting reasoning. *Process-oriented* analysis elicits reasoning but does not construct a thought graph for analysis. *Mutelier* conducts analysis over a structured knowing–doing graph.

Evaluation Protocol. Both human experts and models assess intent–idea and idea–artifact alignment using shared metrics. The *Intent-Idea* alignment rate (I–I) measures the percentage of sub-ideas in a reasoning trace that are consistent with the input intent, where ideas labeled as *preserve* or *elaborate* are treated as intent-aligned. The *Idea-Artifact* alignment rate (I–A) measures the percentage of articulated ideas that are actually reflected in the generated artifact. In Experiment 1, agreement between model predictions and human annotations is evaluated using Pearson correlation (PC) and mean absolute error (MAE) on continuous rates, together with non-redundancy and false positive rate (FPR), where the most-agreed human annotations are treated as ground truth. In Experiment 2, we directly analyze model behavior by reporting I–I, I–A, and the *Intent-Artifact* alignment rate (I–I–A), enabling structural assessment based on the induced thought graphs.

Human Experts and Evaluation Models. For Experiment (1), we recruited 15 music-domain experts (9 male, 6 female; aged 28–47, M=35.4) with backgrounds in composition, music theory, and music education. Experts received rubric training and evaluated fixed prompts and model outputs to avoid confounds. Each sample was independently rated by 10 experts, yielding 4,500 total ratings, with good to excellent inter-rater reliability, as measured by the intra-class correlation coefficient (ICC > 0.8). For Experiment (2), we evaluate 15 LLMs spanning open-source and proprietary families, including *Olmo2-7B, 13B* [OLMO Team, 2024], *Qwen2.5-7B, 32B, 72B* [Yang *et al.*, 2024], *LLaMA3-8B, 72B* [Grattafiori *et al.*, 2024], *Mistral-7B* [Mistral, 2025], *Phi-4 14B* [Abdin *et al.*, 2024], *DeepSeek-R1-70B, V3-671B* [Liu *et al.*, 2024], *Claude-4* [Claude, 2025], *Gemini 1.5-Pro* [Gemini Team, 2024], and *GPT-4o, GPT-o3* [OpenAI, 2025a]. All models use standardized prompting templates to ensure comparability.

User Study. For Experiment (3), we conducted a within-subject, counterbalanced user study with 180 composition enthusiasts familiar with LLM-based tools. Participants sequentially used three AI music creation systems over six weeks (two weeks per system): an output-oriented baseline producing only final outputs, a process-oriented system, and Mutelier, which allows inspection and editing of individual ideas. Across conditions, participants performed identical music creation tasks. After each period, they rated system support for the knowing–doing cycle using 7-point Likert scales measuring transparency (visibility and traceability of reasoning), controllability (ability to intervene in and adjust generation), and collaboration (perceived sense of human–AI co-creation).

Implementation Details. Across all experiments, we use all-MiniLM-L6-v2 as the sentence embedding model [Reimers and Gurevych, 2019; Wang *et al.*, 2020a]. Symbolic music is represented in ABC notation [Yuan *et al.*, 2024]. Intent–idea relations are categorized as preserve (ideas that directly follow the intent), elaborate (ideas that extend the intent), shift (ideas that partially deviate from the intent), or diverge (ideas that depart from the intent). All LLM prompts use fixed templates across experiments. Open-source models (Olmo2, Qwen2.5, LLaMA3, Mistral, Phi, and DeepSeek-R1) are deployed locally using Ollama, while API-based models, including GPT-4o, GPT-o3, DeepSeek-V3-671B, Claude-4, and Gemini-1.5-Pro, are accessed via standard APIs and usage quotas.

4.2 Experiment 1: Validating Mutelier Against Human Experts

Main Results

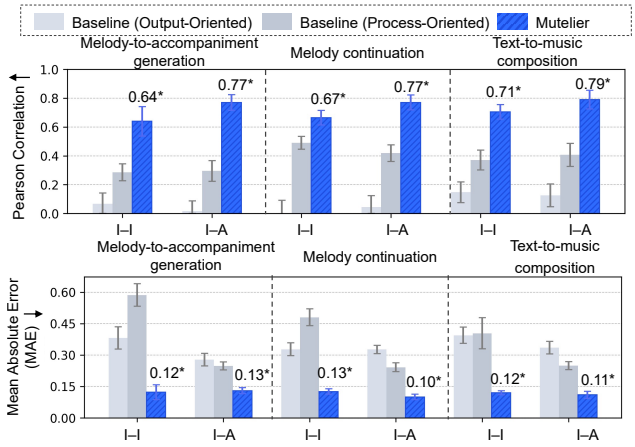


Figure 2: Agreement between different analysis methods and human experts across three tasks. Results are reported using Pearson correlation and mean absolute error for Intent–Idea (I–I) and Idea–Artifact (I–A) alignment. * indicates statistically significant improvement over the process-oriented method at the 0.01 level.

Figure 2 reports agreement between different analysis methods and human experts on the same LLM-generated reasoning traces and musical outputs. Across all three tasks, Mutelier consistently achieves stronger alignment with expert judgments than both baselines on Intent–Idea (I–I) and Idea–Artifact (I–A) dimensions, yielding stable Pearson correlations together with low MAE. In contrast, the Output-oriented baseline ignores how ideas are formed, while the Process-oriented baseline relies on unstructured reasoning traces that remain noisy without grounding in the output. This result validates Mutelier as a reliable process-level analysis framework for artistic creation.

Ablation Analysis

ID	Components			I–I		I–A	
	Structured	Pruning	Verification	PC↑	MAE↓	PC↑	MAE↓
1	✗	✗	✗	0.28	0.38	0.39	0.25
2	✓	✗	✗	0.46	0.25	0.46	0.18
3	✓	✓	✗	0.49	0.17	0.48	0.15
4	✓	✓	✓	**0.67**	**0.12**	**0.78**	**0.11**

Table 1: Ablation analysis of Mutelier. We report Pearson correlation (PC) and mean absolute error (MAE) for I–I and I–A. "Structured" denotes structured thought representation, "Pruning" denotes adaptive redundancy removal, and "Verification" denotes realization checking between articulated ideas and the generated artifact.

Table 1 analyzes the contribution of individual components in Mutelier by progressively enabling structured representation, adaptive pruning, and realization verification. Introducing structured representation substantially improves both I–I and I–A alignment by decomposing raw reasoning traces into explicit ideas, yielding more stable correspondence with human annotations. Adaptive pruning further reduces noise by removing redundant or semantically overlapping ideas, leading to consistent reductions in MAE for both alignment stages. The largest performance gain arises from realization verification, which explicitly checks whether articulated ideas are reflected in the generated artifact. Together, these components enable Mutelier to achieve accurate and robust process-level alignment assessment, closely matching expert judgments across tasks.

Pruning Strategy	Time (s)↓	I-I Non-Redundancy↑
Similarity-based Heuristic Only	5.34	0.76
LLM Only (All-at-once)	17.63	0.71
LLM Only (Pair-wise)	244.86	0.94
Adaptive Pruning†	11.51	0.94

Table 2: Efficiency–accuracy tradeoff of pruning verification strategies. Accuracy is measured against human annotations. † denotes a balanced strategy achieving high accuracy with reduced runtime.

Judging Strategy	I-A Accuracy↑	I-A FPR↓
No QA	0.79	0.93
QA	0.77	0.22
QA + Evidence Triage †	0.86	0.26

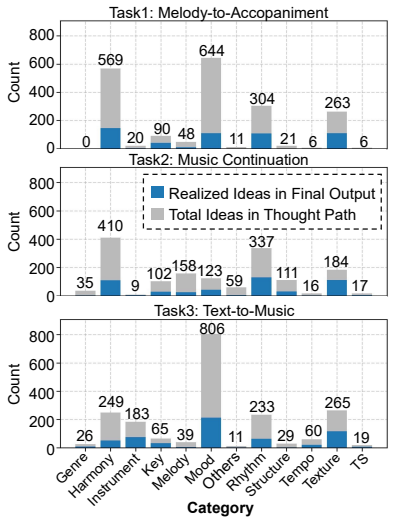
Table 3: Ablation of idea-artifact realization judging strategies. Accuracy and false positive rate (FPR) evaluate correctness and over-claiming of realized ideas; † denotes the best balance.

Efficiency–Accuracy Tradeoff. This experiment is conducted on a subset of samples with human annotations for idea deduplication, where human judgments are treated as ground truth for whether multiple sub-ideas express the same underlying creative idea. Table 2 shows that similarity-based filtering alone is computationally efficient but limited in accuracy, as it relies on heuristic cosine similarity without explicit semantic validation. LLM-based verification substantially improves deduplication accuracy; however, exhaustive pair-wise verification is prohibitively expensive, while all-at-once verification suffers from reduced reliability due to contextual interference. By combining lightweight similarity screening with selective LLM-based verification, adaptive pruning achieves the same accuracy as pair-wise verification with an order-of-magnitude reduction in runtime.

Verification Strategy Ablation. This experiment is conducted on a subset of samples with human annotations indicating whether each articulated idea is realized in the final artifact. Table 3 shows that direct LLM judgment without QA exhibits a severe confirmation bias, achieving moderate accuracy but an extremely high false positive rate, as the model tends to affirm realization even in the absence of supporting evidence. Introducing QA-based judgment substantially reduces false positives by decoupling idea interpretation from artifact inspection. Further adding evidence triage improves overall accuracy while keeping the false positive rate low, demonstrating that verification is essential for reliably distinguishing articulated knowledge from realized outcomes.

Model	I–I (%)	I–A (%)	I–I–A (%)
Mistral-7B	80.3	21.3	14.0
Olmo2-7B	81.6	16.8	12.4
Qwen2.5-7B	82.1	23.4	14.8
Llama3-8B	84.7	22.0	14.3
Olmo2-13B	85.9	18.7	12.7
Phi4-14B	88.4	33.7	23.3
Qwen2.5-32B	86.8	28.3	20.0
Deepseek-R1-70B	89.1	30.7	21.6
Llama3-72B	<u>92.5</u>	31.6	24.0
Qwen2.5-72B	90.7	38.4	28.1
Claude-sonnet-4	**93.8**	**50.3**	**41.9**
Deepseek-V3-671B	91.9	40.8	29.8
Gemini-1.5-pro	91.8	41.4	31.7
GPT-4o	92.4	42.9	36.6
GPT-o3	**93.8**	<u>47.1</u>	<u>38.2</u>

Table 4: Alignment rates across 15 LLMs. I–I, I–A, and I–I–A measure intent–idea, idea–artifact, and intent-artifact alignment rate. Bold and <u>underline</u> denote best and second best.



(a) Histogram of total ideas in the reasoning trace and those realized in the final music across three tasks.

Category	I-A
Others	11.10%
Melody	15.90%
Genre	18.00%
Structure	21.10%
Mood	23.20%
Harmony	24.80%
Time Signature	28.60%
Tempo	30.50%
Rhythm	34.60%
Instrument	38.70%
Key	38.90%
Texture	47.80%
Overall	**29.3%**

(b) I-A realization rates by conceptual category.

Figure 3: Idea-to-Artifact realization diagnostics produced by Mutelier. (a) Distribution of proposed versus realized ideas in the reasoning across tasks. (b) I-A rate broken down by concept type.

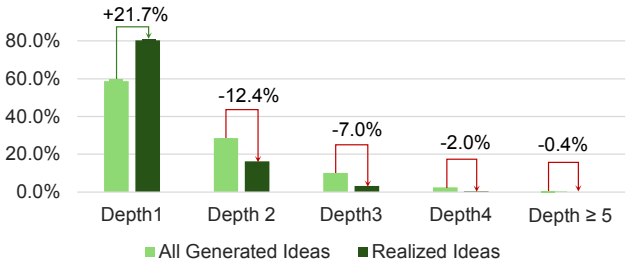


Figure 4: Depth-wise distribution of ideas before and after realization, showing that while ideas are generated across multiple hierarchical levels, realization is strongly biased toward shallow levels, with deeper ideas rarely instantiated.

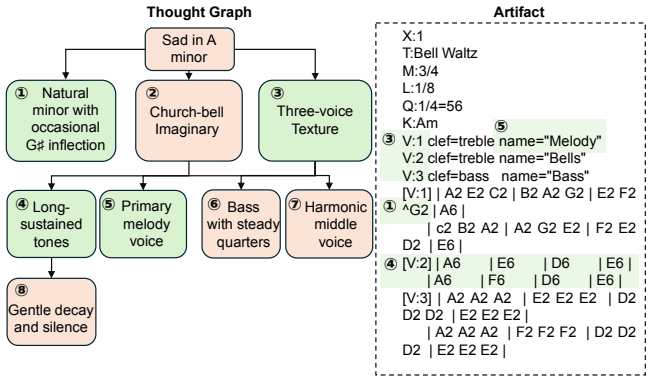


Figure 5: Example of shallow idea realization. Green indicates realized ideas and red indicates unrealized ones; numbers mark intended correspondences between the thought graph and the artifact.

4.3 Experiment 2: Diagnosing Knowing–Doing Gaps Across 15 LLMs

Model-wise Performance

Table 4 reveals a consistent knowing–doing gap across models. While I–I scores are uniformly high, indicating that current LLMs can generally understand and articulate intent-aligned ideas, execution-related metrics remain substantially lower. In particular, the I–I–A metric shows that many intent-aligned ideas fail to be realized in the final artifact, directly reflecting a knowing–doing gap with respect to user intent. At the same time, the lower I–A scores indicate that even ideas articulated by the model itself are frequently not realized, revealing a broader self-execution gap in structured music composition. Model scale consistently improves both I–A and I–I–A, but even the strongest models fall short of full alignment, suggesting that increased scale mitigates but does not eliminate execution failures.

Idea-to-Artifact Realization Diagnostics

Figures 3 and 4 further examine how ideas expressed in reasoning traces of current LLMs are translated into realized musical structure. These figures reveal the following patterns:

Insight 1: Ideation is concentrated in a narrow conceptual space. As shown in Figure 3(a), models repeatedly propose ideas from a small set of categories, while others, such as genre and tempo, are rarely explored. This clustering suggests limited ideational breadth, even before considering realization.

Insight 2: Realization, not ideation, is the primary bottleneck. Figure 3(b) shows that only a minority of proposed ideas are ultimately realized, with substantial variability in realization outcomes. This gap indicates that many ideas remain confined to the reasoning stage and fail to materialize into symbolic music composition.

Insight 3: Realized ideas exhibit shallow elaboration. As illustrated in Figure 4, models generate ideas across multiple depth levels, but execution is strongly biased toward shallow

ideas, with realization probability dropping sharply as depth increases. In Figure 5, although the thought graph spans three hierarchical layers, only the top two are reflected in the realized music. While concept-level ideas such as G# inflection, three-voice texture, and long-sustained tones are realized, descriptive ideas are weakly concretized and inter-idea coordination remains limited. In particular, imagery such as church-bell resonance is not translated into explicit structural actions, nor differentiated across voices, and all parts are rendered with a default piano timbre. Overall, realized ideas exhibit limited multi-step refinement, constraining coherent hierarchical musical planning.

Intent Drift Analysis

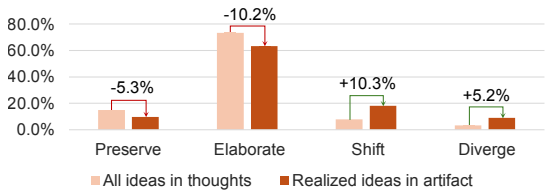


Figure 6: Distribution of intent–idea relations at the thought level and among realized ideas in the final artifact.

Insight 4: Intent drift is introduced during realization through selective idea instantiation. We further analyze how intent–idea relations are transformed from the reasoning trace to the final artifact by categorizing ideas, for each model, into four relation types: preserve, elaborate, shift, and diverge. Figure 6 compares the distribution of intent–idea relations at the thought level with that observed among ideas ultimately realized in the artifact. At the reasoning stage, the majority of ideas align with the user intent through elaboration, indicating that models primarily reason by expanding upon the given intent. However, among realized ideas, the proportion of elaborate decreases substantially, while shift and diverge increase markedly. This redistribution suggests that intent deviation is introduced during the execution process, rather than originating from the reasoning stage itself. A plausible explanation is that reasoning favors intent expansion, while realization is constrained by coherence, leading to selective instantiation that favors shifted or simplified ideas.

Stability of Realization under Fixed Reasoning

Insight 5: Realization remains unstable even under fixed reasoning. Based on the same example in Figure 5, we generate artifacts using an identical intent and reasoning trace, and observe substantial variation in idea realization across 11 repeated runs. Structurally grounded ideas (e.g., three-voice texture) are consistently realized, whereas expressive or stylistic ideas are highly unstable: church-bell imagery and natural minor with G# inflection appear in only 9% and 18% of samples, respectively. This indicates a lack of enforceable mechanisms that bind high-level intentions to stable execution, resulting in selective and inconsistent realization.

System	Transparency ↑	Controllability ↑	Collaboration ↑
Output-oriented	3.58	3.45	4.06
Process-oriented	4.57	3.75	4.57
Mutelier	**5.18**	**4.80**	**5.25**

Table 5: User study results across Output-oriented, Process-oriented, and Mutelier (scale: 1–7).

4.4 Experiment 3: Exposing the Knowing–Doing Process via a Large-Scale User Study

Table 5 shows Mutelier significantly outperforms output-oriented and process-oriented baselines in transparency, controllability, and collaboration ($p < 0.01$). While transparency alone improves understanding, it does not restore user agency. In contrast, Mutelier provides an explicit, editable representation of model reasoning as a thought graph, through which users can adjust the interpretation of articulated ideas without altering internal model reasoning, influencing how intent and ideas are analyzed and reflected in generated artifacts. These results show that effective human–AI co-creation requires not just visibility into reasoning, but shared control over the knowing–doing cycle.

4.5 Limitations

While Mutelier shows strong alignment with human evaluations, its performance depends on the symbolic music understanding of the underlying LLM used for analysis. Stronger models consistently yield higher correlation and lower error. For example, GPT-4o achieves an average PC of 0.67 with an MAE of 0.12, compared to 0.50 / 0.21 for LLaMA3-72B and 0.42 / 0.28 for LLaMA3-8B, indicating that weaker models can limit the performance. As more capable models become accessible, this limitation is expected to diminish. In addition, Mutelier analyzes reasoning traces that are explicitly produced by models; internal latent processes that are not externalized into text remain inaccessible, and thus certain internal aspects of the knowing–doing gap may go unobserved.

5 Conclusion

This paper investigates the knowing–doing gap in LLM-based artistic creation, where models articulate coherent understanding during reasoning but fail to consistently realize it in generated artifacts. We characterize this gap as a realizational failure localized at the reasoning-to-artifact transition and introduce Mutelier, a process-level analytical framework that traces how reasoning-stage ideas are selectively realized, weakened, or lost in the final artifact. Across multiple experiments, we show that Mutelier reveals systematic realization failures, distinguishes knowing–doing behaviors across models, and improves human understanding when the generation process is exposed. Overall, this work shifts the analysis of machine creativity from outcome-level evaluation to process-level realization dynamics, providing a foundation for studying and addressing knowing–doing gaps in creative AI systems. Future work will explore integrating process-level analysis into training and interactive generation loops, and extending the framework to other creative domains and modalities.

Acknowledgments

This work is supported by the project Towards Next-generation Artificial Auditory System with Brain-inspired Technologies (C5052-23G).

References

[Abdin *et al.*, 2024] Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*, 2024.

[Bian *et al.*, 2023] Weizhen Bian, Yijin Song, Nianzhen Gu, Tin Yan Chan, Tsz To Lo, Tsun Sun Li, King Chak Wong, Wei Xue, and Roberto Alonso Trillo. Momusic: A motion-driven human-ai collaborative music composition and performing system. In *AAAI*, volume 37, pages 16057–16062, 2023.

[Chakrabarty *et al.*, 2024] Tuhin Chakrabarty, Philippe Laban, Divyansh Agarwal, Smaranda Muresan, and Chien-Sheng Wu. Art or artifice? large language models and the false promise of creativity. In *CHI*, pages 1–34, 2024.

[Chen *et al.*, 2024] Dongping Chen, Ruoxi Chen, Shilin Zhang, Yaochen Wang, Yinuo Liu, Huichi Zhou, Qihui Zhang, Yao Wan, Pan Zhou, and Lichao Sun. Mllm-as-a-judge: Assessing multimodal llm-as-a-judge with vision-language benchmark. In *ICML*, 2024.

[Cideron *et al.*, 2024] Geoffrey Cideron, Sertan Girgin, Mauro Verzetti, Damien Vincent, Matej Kastelic, Zalán Borsos, Brian McWilliams, Victor Ungureanu, Olivier Bachem, Olivier Pietquin, et al. Musicrl: Aligning music generation to human preferences. In *ICML*, pages 8968–8984, 2024.

[Claude, 2025] Claude. Official website. https://claude.ai/, 2025. Accessed: 2025-12-01.

[Crisp, 2014] Roger Crisp. *Aristotle: nicomachean ethics*. Cambridge University Press, 2014.

[Desmond *et al.*, 2025] Michael Desmond, Zahra Ashktorab, Werner Geyer, Elizabeth M Daly, Martin Santillan Cooper, Qian Pan, Rahul Nair, Nico Wagner, and Tejaswini Pedapati. Evalassist: Llm-as-a-judge simplified. In *AAAI*, volume 39, 2025.

[D'Souza *et al.*, 2025] Jennifer D'Souza, Hamed Babaei Giglou, and Quentin Münch. Yescieval: Robust llm-as-a-judge for scientific question answering. In *ACL*, pages 13749–13783, 2025.

[Franceschelli and Musolesi, 2024] Giorgio Franceschelli and Mirco Musolesi. Creativity and machine learning: A survey. *ACM Computing Surveys*, 56(11):1–41, 2024.

[Friedman and Dieng, 2023] Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023.

[Gemini Team, 2024] Gemini Team. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[Grattafiori *et al.*, 2024] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[Hu *et al.*, 2023] Zhejing Hu, Xiao Ma, Yan Liu, Gong Chen, Yongxu Liu, and Roger B Dannenberg. The beauty of repetition: an algorithmic composition model with motif-level repetition generator and outline-to-music generator in symbolic music generation. *IEEE Transactions on Multimedia*, 26:4320–4333, 2023.

[Hu *et al.*, 2024] Zhejing Hu, Yan Liu, Gong Chen, Xiao Ma, Shenghua Zhong, and Qianwen Luo. Responding to the call: Exploring automatic music composition using a knowledge-enhanced model. In *AAAI*, volume 38, pages 521–529, 2024.

[Hu *et al.*, 2025a] Zhejing Hu, Yan Liu, Gong Chen, and Bruce XB Yu. Complex: music theory lexicon constructed by autonomous agents for automatic music generation. In *IJCAI*, pages 7419–7427, 2025.

[Hu *et al.*, 2025b] Zhejing Hu, Yan Liu, Gong Chen, and Bruce XB Yu. Compose with me: Collaborative music inpainter for symbolic music infilling. In *AAAI*, volume 39, pages 1327–1335, 2025.

[Huang *et al.*, 2025] Zhongzhan Huang, Shanshan Zhong, Pan Zhou, Shanghua Gao, Marinka Zitnik, and Liang Lin. A causality-aware paradigm for evaluating creativity of multimodal large language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[Liu *et al.*, 2024] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[Melechovský *et al.*, 2024] Jan Melechovský, Abhinaba Roy, and Dorien Herremans. Midicaps: A large-scale MIDI dataset with text captions. In *ISMIR*, pages 858–865, 2024.

[Meng *et al.*, 2025] Siwei Meng, Yawei Luo, and Ping Liu. Grounding creativity in physics: A brief survey of physical priors in AIGC. In *IJCAI*, pages 10593–10602, 2025.

[Mistral, 2025] Mistral. Official website. https://mistral.ai/, 2025. Accessed: 2025-12-01.

[OLMO Team, 2024] OLMO Team. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.

[OpenAI, 2025a] OpenAI. Gpt-4o. https://chatgpt.com/, 2025. Accessed: 2025-08-02.

[OpenAI, 2025b] OpenAI. Gpt-o3. https://chatgpt.com/, 2025. Accessed: 2025-08-02.

[Panickssery *et al.*, 2024] Arjun Panickssery, Samuel Bowman, and Shi Feng. Llm evaluators recognize and favor their own generations. *NeurIPS*, 37:68772–68802, 2024.

[Peng, 2022] Nanyun (Violet) Peng. Controllable text generation for open-domain creativity and fairness. In *IJCAI*, pages 5821–5825, 2022.

[Reimers and Gurevych, 2019] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP*, pages 3980–3990, 2019.

[Wan *et al.*, 2024] Qian Wan, Siying Hu, Yu Zhang, Piaohong Wang, Bo Wen, and Zhicong Lu. "it felt like having a second mind": Investigating human-ai co-creativity in prewriting with large language models. *Proc. ACM Hum. Comput. Interact.*, 8:1–26, 2024.

[Wang *et al.*, 2020a] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *NeurIPS*, 33:5776–5788, 2020.

[Wang *et al.*, 2020b] Ziyu Wang, Ke Chen, Junyan Jiang, Yiyi Zhang, Maoran Xu, Shuqi Dai, and Gus Xia. POP909: A pop-song dataset for music arrangement generation. In *ISMIR*, pages 38–45, 2020.

[Wang *et al.*, 2021] Songhe Wang, Zheng Bao, and Jingtong E. Armor: A benchmark for meta-evaluation of artificial music. In *ACM Multimedia*, pages 5583–5590, 2021.

[Wang *et al.*, 2023] Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. In *NeurIPS*, 2023.

[Wang *et al.*, 2025a] Yashan Wang, Shangda Wu, Jianhuai Hu, Xingjian Du, Yueqi Peng, Yongxin Huang, Shuai Fan, Xiaobing Li, Feng Yu, and Maosong Sun. Notagen: Advancing musicality in symbolic music generation with large language model training paradigms. In *IJCAI*, pages 10207–10215, 2025.

[Wang *et al.*, 2025b] Zihao Wang, Le Ma, Yuhang Jin, Yongsheng Feng, Xin Pan, Shulei Ji, and Kejun Zhang. Ai-assisted human-pet artistic musical co-creation for wellness therapy. In *IJCAI*, pages 10216–10224, 2025.

[Wei *et al.*, 2022] Jason Wei, Xuezhi Wang, Dale Schuurmans, and et al. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

[Yang *et al.*, 2024] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[Ye *et al.*, 2025] Ziyi Ye, Xiangsheng Li, Qiuchi Li, Qingyao Ai, Yujia Zhou, Wei Shen, Dong Yan, and Yiqun Liu. Learning llm-as-a-judge for preference alignment. In *ICLR*, 2025.

[Yu *et al.*, 2025] Qingchen Yu, Zifan Zheng, Shichao Song, Feiyu Xiong, Bo Tang, Ding Chen, et al. xfinder: Large language models as automated evaluators for reliable evaluation. In *ICLR*, 2025.

[Yuan *et al.*, 2023] Ruibin Yuan, Yinghao Ma, Yizhi Li, Ge Zhang, Xingran Chen, Hanzhi Yin, Yiqi Liu, Jiawen Huang, Zeyue Tian, Binyue Deng, et al. Marble: Music audio representation benchmark for universal evaluation. *NeurIPS*, 36:39626–39647, 2023.

[Yuan *et al.*, 2024] Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou, et al. Chatmusician: Understanding and generating music intrinsically with llm. In *ACL (Findings)*, 2024.

[Zhang *et al.*, 2024] Jifan Zhang, Lalit Jain, Yang Guo, Jiayi Chen, Kuan Zhou, Siddharth Suresh, Andrew Wagenmaker, Scott Sievert, Timothy T Rogers, Kevin G Jamieson, et al. Humor in ai: Massive scale crowdsourced preferences and benchmarks for cartoon captioning. *NeurIPS*, 37:125264–125286, 2024.

[Zhang *et al.*, 2025] Qiyuan Zhang, Yufei Wang, Tiezheng Yu, Yuxin Jiang, Chuhan Wu, Liangyou Li, Yasheng Wang, Xin Jiang, Lifeng Shang, Ruiming Tang, Fuyuan Lyu, and Chen Ma. Reviseval: Improving llm-as-a-judge via response-adapted references. In *ICLR*, 2025.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *NeurIPS*, 36:46595–46623, 2023.

[Zhong *et al.*, 2024] Shanshan Zhong, Zhongzhan Huang, Shanghua Gao, Wushao Wen, Liang Lin, Marinka Zitnik, and Pan Zhou. Let's think outside the box: Exploring leap-of-thought in large language models with creative humor generation. In *CVPR*, pages 13246–13257, 2024.

[Zhou *et al.*, 2024] Ziya Zhou, Yuhang Wu, Zhiyue Wu, Xinyue Zhang, Ruibin Yuan, Yinghao Ma, Lu Wang, Emmanouil Benetos, Wei Xue, and Yike Guo. Can llms "reason" in music? an evaluation of llms' capability of music understanding and generation. In *ISMIR*, pages 103–110, 2024.