Audio-Side Reinforcement Learning for Symbolic Music Style Adaptation: Extending SMART with Composite Rewards

Abdolrahim Tooranian
University of Pisa
a.tooranian@studenti.unipi.it

Bob L. T. Sturm
KTH Royal Institute of Technology
bobs@kth.se

Abstract

Controlling the stylistic behavior of symbolic music generation models remains difficult. This paper proposes an audio-reward fine-tuning framework that bridges editable symbolic generation with audio-domain models. The framework renders generated MIDI to audio and optimizes the symbolic policy with Group Relative Policy Optimization (GRPO) using a composite reward that can combine prompt alignment, learned musical-quality estimation, and reference-audio similarity. We apply the framework to MIDI-LLM and evaluate it through focused style experiments and a listening test. The results show that our framework can be used to steer outputs toward target styles in focused style-adaptation settings, with listeners generally rating tuned models as closer to the target style. However, the experiments also reveal limitations: reward optimization is very sensitive to prompt distribution and reward configuration, and can lead to training instability or reduced diversity.

1 INTRODUCTION

Symbolic music generation has become an important direction in generative music research because it produces structured musical representations that can be edited, arranged, and reused in creative workflows among other benefits. Recent symbolic music generation models can produce musically well-formed outputs, but controlling style, mood, or musical character from text is still difficult (Bhandari et al., 2025; Wu et al., 2025a). Datasets such as MidiCaps and MetaScore have improved the amount of text-conditioned symbolic music data, but symbolic datasets paired with rich natural-language descriptions are still limited compared with the audio domain (Melechovsky et al., 2024; Xu et al., 2025). Prior work has also suggested that symbolic music generation may lag behind audio-domain generation in part because symbolic training data is scarcer (Chen et al., 2024). Because of this, standard next-token training can produce reasonable outputs without giving strong control over finer stylistic details.

At the same time, the audio domain already has a richer set of models that can evaluate generated music in useful ways. Contrastive text-audio models such as CLAP and MuLan can measure alignment between music and natural-language descriptions (Elizalde et al., 2023; Huang et al., 2022). Learned evaluators such as AudioBox Aesthetics and SongEval can estimate qualities such as enjoyment, coherence, and musicality without using a reference track (Tjandra et al., 2025; Yao et al., 2025). These models do not directly solve symbolic generation, but they make it possible to score generated music after rendering to audio. This creates a practical bridge: symbolic models can be trained with feedback coming from the audio side.

Reinforcement learning is a natural way to use this kind of feedback. Instead of requiring supervision directly in symbolic token space, reinforcement learning allows a symbolic generator to optimize against black-box reward signals computed after rendering. More broadly, recent work has shown that reinforcement learning and preference-based optimization can improve generative models (Cideron et al., 2024; Wang et al., 2025). SMART showed that a symbolic music generator can be fine-tuned using an aesthetic reward computed from rendered audio outputs (Jonason et al., 2025). This gives a clear starting point for using audio-based evaluators to guide symbolic music generation.

In this paper, we study reinforcement-learning fine-tuning of MIDI-LLM (Wu et al., 2025a) using rewards computed after rendering generated MIDI to audio. Our goal is not to replace symbolic generation with direct audio generation, but to keep the structural benefits of symbolic modeling while using stronger supervision from the audio domain. We focus on three reward sources: text-audio alignment, learned preference estimates of coherence and musicality, and audio–audio similarity to reference material. These rewards are combined in a GRPO-based optimization framework (Shao et al., 2024) to fine-tune the symbolic policy.

We evaluate the method through qualitative examples, automatic metrics, and a human listening test. First, we study how different reward combinations affect generated outputs in focused style settings. We then use piano blues as a case study and compare training checkpoints and reward settings in a listening test where participants judge stylistic match to a reference recording. Finally, we test whether the approach extends beyond a single style by fine-tuning on a small set of prompts from related genres such as jazz, lo-fi, and soul.

Our results show that text-alignment rewards can strengthen style-related cues, but can hurt musicality when used alone. Adding preference rewards leads to more coherent and convincing generations. We also identify a reward–diversity trade-off. In some experiments, reward increased while the generated samples converged toward a narrow set of similar outputs. Training also became unstable when extended to broader prompt distributions and more diverse stylistic regimes.

2 BACKGROUND AND RELATED WORK

2.1 Text-conditioned symbolic music generation

Symbolic music generation has focused on autoregressive modeling of event-based note sequences, where Transformers and music-specific tokenizations improved long-range musical structure and rhythmic regularity (Huang et al., 2019; Huang and Yang, 2020; Thickstun et al., 2024). More recently, attention has shifted toward *text-conditioned* symbolic generation, where the goal is not only to produce well-formed MIDI but also to match natural-language descriptions of style, instrumentation, mood, or other musical attributes.

Progress in this direction has been enabled in part by new datasets that pair symbolic music with textual descriptions or rich metadata. MidiCaps (Melechovsky et al., 2024) introduces a large-scale dataset of MIDI-caption pairs, while MetaScore (Xu et al., 2025) provides symbolic music paired with extensive metadata and LLM-enhanced pseudo-captions. Building on such resources, recent systems such as *text2midi* (Bhandari et al., 2025) and *MIDI-LLM* (Wu et al., 2025a) demonstrate end-to-end text-to-MIDI generation from free-form prompts. However, despite these advances, controllable symbolic music generation remains challenging: prompt-conditioned models often capture broad cues while still struggling with finer stylistic fidelity and higher-level musical qualities.

2.2 Audio-side embedding and evaluation models

Contrastive text-audio models such as *CLAP* (Elizalde et al., 2023), *MuLan* (Huang et al., 2022), and *MuQ-MuLan* (Zhu et al., 2025) learn shared representation spaces in which natural-language descriptions and audio clips can be compared directly. In this setting, text-audio similarity can be used as a proxy for prompt adherence, making these models useful for retrieval, zero-shot tagging, automatic evaluation, and reward design in language-conditioned music generation.

Self-supervised music representation models such as *MERT* and *MuQ* learn transferable audio embeddings from large collections of unlabeled music audio (Li et al., 2024; Zhu et al., 2025). In our setting, such embeddings are important because they provide the audio representations used by higher-level predictors and reward models.

Additionally, recent work has introduced learned *audio-side* predictors of perceptual quality and aesthetics. *Meta AudioBox Aesthetics* predicts several reference-free audio quality dimensions from short clips, while *SongEval* provides full-song aesthetic ratings and trained predictors for dimensions such as coherence and overall musicality (Tjandra et al., 2025; Yao et al., 2025).

2.3 Preference optimization and reinforcement learning for music generation

Standard next-token training does not directly optimize the sequence-level properties that matter most at inference time, such as stylistic match, coherence, or listener preference. This has motivated a growing line of work on preference optimization and reinforcement-learning-based post-training for music generation. In the audio domain, *MusicRL* fine-tunes a text-to-music model using reward models and large-scale human preference data (Cideron et al., 2024). In the symbolic domain, *NotaGen* applies preference-style alignment to improve musicality and controllability in symbolic generation (Wang et al., 2025).

At the same time, optimizing learned or proxy rewards can introduce failure modes. In reinforcement learning, reward hacking or specification gaming occurs when a model exploits weaknesses in the reward signal rather than improving the intended behavior (Amodei et al., 2016); under strong optimization pressure, this can become a Goodhart-style failure where the proxy objective diverges from the underlying goal (Manheim and Garrabrant, 2018; Gao et al., 2023). In creative domains such as music, this can appear as reduced diversity, repetitive outputs, or improvements in automatic reward scores that do not correspond to better musical results (Kirk et al., 2024; Jonason et al., 2025; Wu et al., 2025b). These risks are especially relevant when using embedding-based or aesthetic predictors as rewards, since such models are only imperfect proxies for human musical judgment.

We build our framework on *SMART* (Jonason et al., 2025), which fine-tunes a symbolic music generator with reinforcement learning using an *audio-domain* aesthetic reward computed after rendering symbolic outputs to audio. This result is important because it suggests that symbolic generators can be steered using perceptual reward signals that are unavailable in the symbolic domain alone. Our work builds on this general idea, but focuses specifically on *style steering* and studies a broader combination of reward sources, including text-audio alignment, learned preference estimates of musical quality, and optional reference-based audio similarity.

3 METHOD

Figure 1 shows our reinforcement-learning framework for steering symbolic music generation using rewards computed after rendering generated MIDI to audio. The key idea is to keep generation in the symbolic domain, where MIDI-like representations remain compact and editable, while using audio-domain models to evaluate the generated music. For each prompt, the symbolic policy generates a group of candidate MIDI sequences, which are decoded and synthesized to audio. The rendered audio is then scored by several reward components, such as text-audio alignment, learned aesthetic prediction, and audio-audio reference similarity. These components are normalized and combined into a scalar reward, converted into group-relative advantages, and used to update the policy with GRPO while regularizing it toward a fixed reference policy. We apply this framework to MIDI-LLM (Wu et al., 2025a), using GRPO as the optimization method and a composite reward based on text-audio alignment, learned aesthetic prediction, audio–audio reference similarity.

3.1 Symbolic generation as sequence-level reinforcement learning

We treat text-conditioned MIDI generation as a sequence-level reinforcement-learning problem. Given a prompt q, the policy $\pi_\theta(\cdot \mid q)$ generates a complete MIDI-token sequence o, which is decoded to MIDI and rendered to audio. Rewards are computed after this rendering step, so they are terminal, sparse, and non-differentiable with respect to the symbolic tokens. This allows the policy to optimize audio-domain signals while still producing editable symbolic output. The objective is to improve the

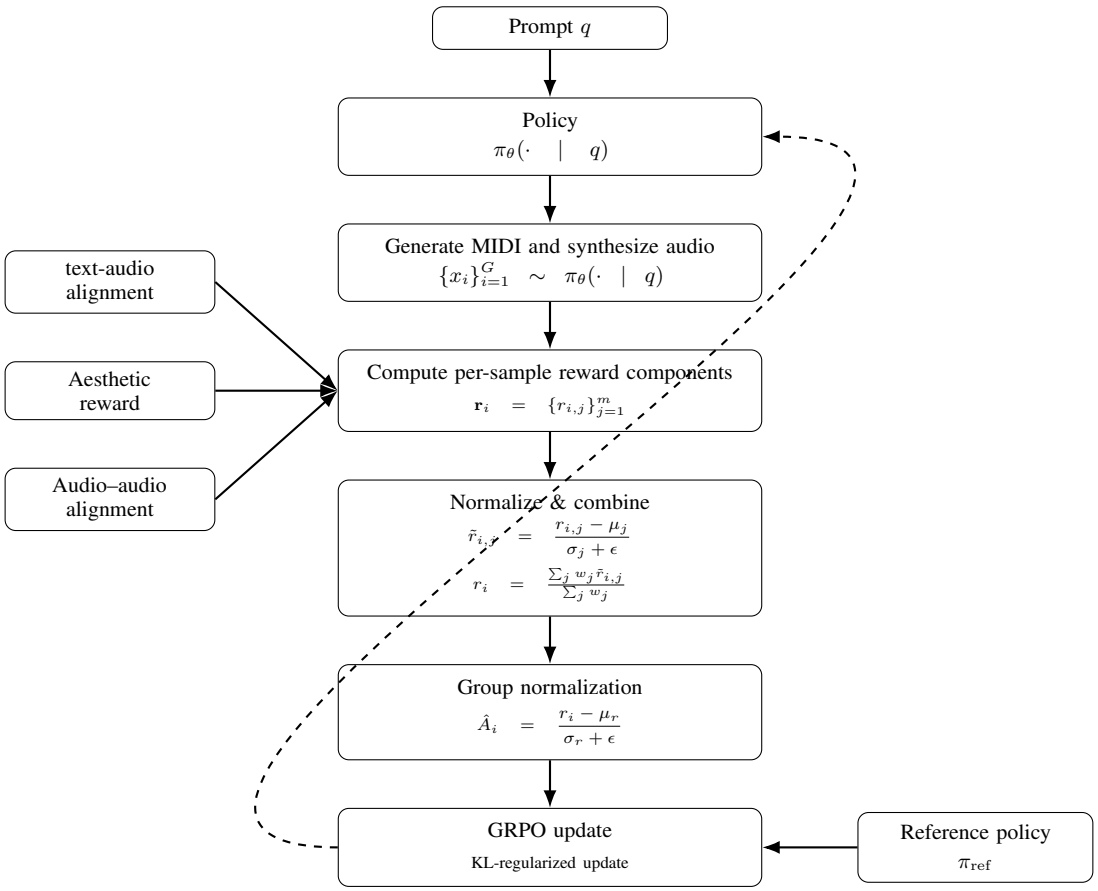

Figure 1: GRPO loop with composite rewards. The policy samples a group of MIDI samples. These samples are synthesized to audio, reward components are computed and combined into a scalar r_i, group-relative advantages $\hat{A}_i$ are formed, and the policy is updated with a KL penalty to a fixed reference policy.

expected terminal reward,

$$J(\theta) = \mathbb{E}_q \mathbb{E}_{o \sim \pi_\theta(\cdot|q)} \left[R(q, o) \right],$$

where $R(q, o)$ is computed from the rendered audio.

3.2 Optimization with GRPO

We optimize the symbolic policy using GRPO. For each prompt q, the current policy samples a group of G candidate completions:

$$\{o_i\}_{i=1}^{G} \sim \pi_\theta(\cdot \mid q).$$

Each completion is decoded into MIDI and rendered into audio: $o_i \longrightarrow \text{MIDI} \longrightarrow x_i$. The rendered audio x_i is then evaluated by the reward functions described below, producing one scalar reward per completion. GRPO compares samples within the same prompt group and updates the policy so that completions with higher group-relative reward become more likely. The full objective formulation is included in Appendix A.1.

In our implementation, the policy is initialized from MIDI-LLM, a text-to-MIDI model based on Llama 3.2 1B and AMT-style MIDI tokens (Meta, 2024; Wu et al., 2025a; Thickstun et al., 2024). The reference policy is the frozen pre-RL MIDI-LLM checkpoint. Generated token sequences are decoded using the AMT representation and rendered with the FluidR3_GM.sf2 General MIDI SoundFont. If decoding or rendering fails, the sample cannot be evaluated by the audio reward models and is assigned a fixed fallback reward.

3.3 Composite reward framework

The framework supports several reward components. Each component evaluates a different property of the rendered audio. In the main instantiation, we use text-audio alignment, learned aesthetic prediction, and audio–audio reference similarity.

The text-audio alignment reward measures whether the generated audio matches the conditioning prompt. In the general framework, let $e_t(\cdot)$ be a text encoder and $e_a(\cdot)$ be an audio encoder mapping text and audio into a shared embedding space. Given prompt q and generated audio x, the reward is cosine similarity:

$$r_{\text{align}}(q, x) = \cos(e_t(q), e_a(x)).$$

This component encourages the generated music to match prompt-level semantic attributes such as genre, instrumentation, mood, or style.

In our implementation, we instantiate e_t and e_a with MuQ-MuLan (Zhu et al., 2025). Thus, the prompt and the rendered audio are embedded into the same text–music representation space, and their cosine similarity is used as the text-alignment reward.

Text alignment alone does not guarantee that the generated result is musically convincing. A sample may match the prompt superficially while still sounding incoherent, repetitive, or weak. To address this, the framework includes a learned aesthetic predictor $f_{\text{pref}}(x) \in \mathbb{R}^d$, which maps audio to one or more scalar preference-related scores.

Each output dimension can be treated as a separate reward component.

In this work, we instantiate this predictor using SongEval (Yao et al., 2025). We use two dimensions: coherence and musicality. For rendered audio x, the corresponding reward components are $r_{\text{coh}}(x)$, and $r_{\text{mus}}(x)$. The coherence score rewards outputs that sound structurally and musically consistent, while the musicality score rewards outputs that are perceived as more convincing or satisfying as music.

The framework can also steer the generator toward a target style or reference set using audio–audio similarity. For a prompt q, let $\mathcal{X}(q)$ denote the set of selected reference recordings. Both generated and reference audio are split into segments, and each segment is embedded using an audio embedding function. Let

$$E(x) = \{e_i(x)\}_{i=1}^{N_x}$$

denote the segment embeddings of an audio sample x. For generated audio $\hat{x}$, we obtain generated segment embeddings

$$\hat{E} = \{\hat{e}_i\}_{i=1}^{N_{\hat{x}}},$$

and for the selected references we form a pooled reference set

$$E_{\text{ref}} = \text{concat}_{x \in \mathcal{X}(q)} E(x) = \{r_j\}_{j=1}^{M}.$$

We compute pairwise cosine similarities between generated and reference segments:

$$S_{i,j} = \cos(\hat{e}_i, r_j).$$

For each generated segment, the similarities to the reference pool are aggregated. In our implementation, we use top-k mean aggregation:

$$u_i = \frac{1}{k'} \sum_{j \in \text{TopK}(S_{i,:}, k')} S_{i,j}, \qquad k' = \min(k, M).$$

The final audio–audio reward is the mean over generated segments:

$$r_{\text{audio}}(\hat{x}; q) = \frac{1}{N_{\hat{x}}} \sum_{i=1}^{N_{\hat{x}}} u_i.$$

We split both generated and reference audio into approximately 30-second chunks. Segment embeddings are extracted with MuQ (Zhu et al., 2025), and cosine similarity is used to compare them. Based on an empirical layer-selection analysis, we use intermediate layers $[4, 7]$ from the 12-layer MuQ model: mid-level layers generally produced similar results, while this pair showed the best qualitative agreement in our listening checks. We then apply top-k aggregation with $k = 2$.

To discourage collapse among completions sampled for the same prompt, the framework optionally includes an intra-group similarity penalty(see Appendix A.2).

3.4 Reward normalization and scalarization

The reward components come from different models and have different empirical scales. Therefore, each component is normalized before scalarization:

$$\tilde{r}_j = \frac{r_j - \mu_j}{\sigma_j + \epsilon},$$

where j indexes reward components, and μ_j and σ_j are estimated from a calibration set. In our implementation, these statistics are computed from a random set of 400 prompts from the SongDescriber (Manco et al., 2023) dataset.

The normalized components are combined into a single scalar reward:

$$R = \frac{\sum_j w_j \tilde{r}_j}{\sum_j w_j}, \qquad w_j \geq 0.$$

where

$$j \in \{\text{align}, \text{coh}, \text{mus}, \text{audio}\}.$$

The weights are set empirically for each experimental condition, reflecting the desired trade-off between prompt adherence, musical quality, reference-style similarity. If MIDI decoding or audio rendering fails, the sample receives a fixed invalid-sample reward, $R_{\text{fail}} = -1.0$. This value was chosen empirically to be lower than the rewards typically assigned to valid samples in our experiments.

4 EXPERIMENTS AND RESULTS

We evaluate whether the proposed reward framework can improve style control in symbolic music generation. The base MIDI-LLM model often shows weak prompt fidelity; for example, for the prompt "blues", the model does not produce a clearly blues-like result Play ▶, and for the prompt "techno", it produces Play ▶. This limitation is consistent with the observation in MIDI-LLM that text embeddings can have limited influence during generation (Wu et al., 2025a). We evaluate the framework through reward-component case studies, held-out automatic metrics on a related family jazz/soul/lo-fi prompts, a piano-blues listening test, and an analysis of training instability. The audio-alignment reference sets consisted of four royalty-free blues tracks from Pixabay, two royalty-free techno tracks from Pixabay, and five public-domain Chopin Prelude recordings: blues Play ▶, techno Play ▶, and Chopin Play ▶.

4.1 Reward component case studies

We first examine how different reward components affect model behavior in focused settings. Using the text-alignment reward alone improves prompt relevance at a broad level, especially for genre and instrumentation cues. For the prompt "blues", fine-tuning with text alignment alone increases the mean text-alignment reward from 0.19 at the beginning of training to 0.41. The resulting output Play ▶ contains some blues-like intervals, suggesting that the reward partially addresses the prompt-fidelity problem. However, the result remains musically limited, indicating that text-audio similarity alone can overemphasize local stylistic cues without preserving overall musical coherence.

Adding aesthetic rewards produces more stable behavior. For the prompt "blues", combining text alignment with coherence and musicality rewards produces Play ▶. This suggests that the aesthetic rewards help stabilize the stylistic pressure induced by text alignment. For the prompt "techno", the same reward combination produces Play ▶. Although the techno output does not fully capture the target style, it sounds more coherent and stylistically closer to the prompt than the base model output.

For more specific styles, the improvement is more subtle. For the prompt "chopin romantic piano classical", the text+aesthetic model produces Play ▶, compared with the base model output Play ▶. The fine-tuned model does not necessarily produce a fully convincing Chopin-style piece,

but it more consistently uses piano instrumentation, while the base model may introduce unrelated instruments.

Finally, we evaluate the audio-alignment reward, which explicitly pulls generations toward a reference embedding corpus. For the Chopin prompt, we use a small subset of Chopin Preludes as the reference set Play ▶ and apply top-2 aggregation over reference segments. Using only the audio-alignment reward produces Play ▶. Combining the audio-alignment reward with aesthetic rewards produces Play ▶. Although the result still does not fully reproduce Chopin's style, it appears closer to the reference style than outputs obtained without the audio-alignment reward. In the blues setting, using Play ▶ as the reference, the mixed reward produces Play ▶. In these focused examples, the reward components appear to play complementary roles: text-alignment increases prompt-relevant cues, aesthetic rewards help maintain musical coherence, and audio-reference alignment can provide an additional reference-specific steering signal.

4.2 Style-specific fine-tuning on related prompts

We next test whether the reward framework transfers beyond single-prompt case studies while remaining within a coherent style family. We use 200 LLM-generated prompts (OpenAI GPT 5.4) describing jazz-adjacent, lo-fi, soul, neo-soul, lounge, and jazzhop instrumentals. The prompts combine style terms with instrumentation and mood descriptors, frequently referring to electric piano or Rhodes, upright bass, brushed drums, clean guitar, soft percussion, and late-night or relaxed atmospheres. This setting is broader than a single prompt, but still narrower than an open-ended genre distribution. We split the prompts into 100 training prompts and 100 held-out evaluation prompts. The model is fine-tuned on the training split for 400 steps using text-alignment and aesthetic rewards. We use LoRA with rank $r = 200$, LoRA alpha $\alpha = 400$, learning rate 4×10^{-6}, KL coefficient $\beta = 0.001$, and reward weights 1.0, 0.08, and 0.08 for text alignment, coherence, and musicality, respectively.

After fine-tuning, we evaluate on the held-out split. For each evaluation prompt, we generate four samples before and after fine-tuning. We then compute CLAP and MuQ-MuLan cosine similarity between each generated sample and its prompt. The same evaluation procedure is applied before fine-tuning, so the comparison uses the same held-out prompt set. For a randomly selected held-out prompt, representative examples are provided before fine-tuning Play ▶ and after LoRA fine-tuning Play ▶.

Because MuQ-MuLan and SongEval are also reward components, these metrics should be interpreted as reward-side diagnostics; CLAP and the listening test provide more independent checks. As shown in Table 1, both text-audio metrics improve after fine-tuning. CLAP similarity increases from 0.114 to 0.204, while MuQ-MuLan similarity increases from 0.036 to 0.187. The medians show the same trend, indicating that the improvement is not driven only by a small number of outliers.

We also compute SongEval scores before and after fine-tuning. As shown in Table 2, all five SongEval dimensions improve, with the overall score increasing from 2.488 to 2.634. The corresponding score distributions are shown in Figure 2. These results suggest that, on a related family of prompts, the reward framework can improve prompt alignment and the learned aesthetic estimates used during training.

4.3 Listening test

We conduct an online listening test to evaluate whether listeners perceive generated clips as matching a target style. The test focuses on "piano blues" to keep the rating task interpretable and manageable. Because participants had to rate multiple reward configurations and training checkpoints, using a single reference-defined style reduced the burden of learning several style targets during the test. Participants first listened to a reference piano-blues clip and then rated generated clips according to how well each clip matched the style of the reference. They were instructed to judge stylistic match rather than exact similarity.

We evaluate two fine-tuned model variants: a *text+aesthetic* model trained with text-alignment, aesthetic rewards and a *text+aesthetic+audio* model that additionally used the audio-reference reward. The two models differ only in whether the audio-reference reward is enabled. Their main reward weights and training settings are shown in Table 3.

The evaluated checkpoints were GRPO training steps 0, 40, 80, 120, and 160, where one step corresponds to one policy update after generating and scoring a group of samples for the training prompt. Since both fine-tuned models originate from the same base MIDI-LLM checkpoint, step 0 is treated as a shared baseline. The test included three generated clips from the shared baseline and three generated clips for each model at each later checkpoint, giving 27 generated clips in total. Two control clips were also included: a positive control taken from the reference audio, and a negative control consisting of a single-note segment. Thus, the test contained 29 stimuli. Representative examples from the listening test are: shared step-0 baseline Play ▶, *text+aesthetic* model at step 160 Play ▶, and *text+aesthetic+audio* model at step 160 Play ▶.

Participants rated each clip on a 10-point scale, where 1 indicates a very poor style match and 10 indicates a very strong style match. Clips were presented in randomized order. Of 32 participants, 24 passed quality control and were retained for analysis. A submission was discarded if the participant rated the reference-derived positive control below 8 or the single-note negative control above 2. These thresholds were chosen to retain participants whose ratings followed the expected direction of the task. The same qualitative trend was also visible before exclusion, although the retained set was used for the reported inferential analysis.

Descriptive statistics are shown in Table 4. The text+aesthetic model improves slowly, remaining close to the shared baseline at steps 40 and 80 before improving at later checkpoints. In contrast, the text+aesthetic+audio model shows earlier and more consistent gains. The highest mean rating among generated conditions is achieved by the text+aesthetic+audio model at step 160 ($M = 7.15$, median $= 7$, $SD = 2.11$). Relative to the shared baseline, the step-160 text+aesthetic+audio model improves by approximately 3.53 rating points, while the text+aesthetic model improves by approximately 2.38 points. Aggregated across the fine-tuned checkpoints 40-160, the text+aesthetic+audio model also receives higher ratings overall ($M = 5.87$, $SD = 2.51$) than the text+aesthetic model ($M = 4.55$, $SD = 2.60$). Figure 3 shows the mean listener ratings across training steps.

Table 1: Audio-text similarity statistics on the held-out 100-prompt evaluation split, before and after LoRA fine-tuning on the separate 100-prompt training split. For each evaluation prompt, four samples were generated.

Metric	Condition	Mean	Median	Std
CLAP	Before fine-tuning	0.114	0.118	0.118
CLAP	After LoRA fine-tuning	0.204	0.208	0.116
MuQ-MuLan	Before fine-tuning	0.036	0.028	0.127
MuQ-MuLan	After LoRA fine-tuning	0.187	0.187	0.135

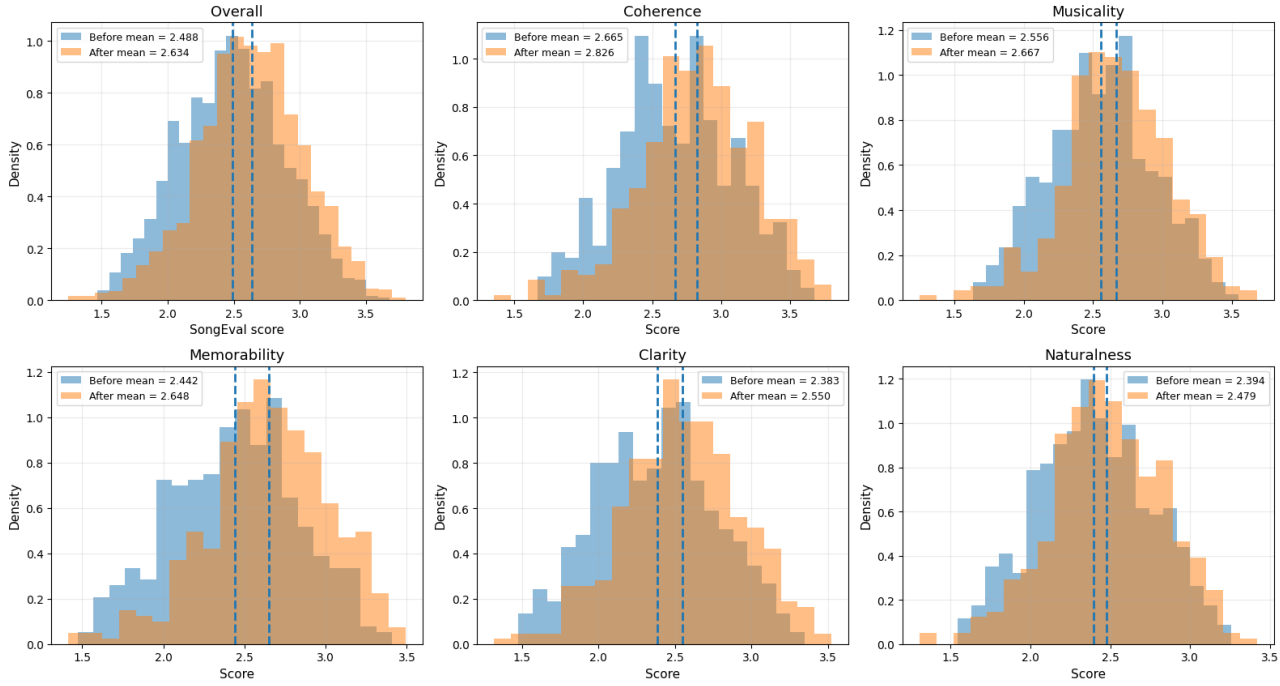

Figure 2: SongEval score distributions before and after LoRA fine-tuning. The figure compares the overall score distribution and the five individual SongEval dimensions.

For inferential analysis, we excluded the shared step-0 baseline and performed a repeated-measures analysis over checkpoints 40, 80, 120, and 160. Ratings were first averaged within each retained submission for each model and checkpoint, so that the three clips per condition were treated as repeated observations rather than independent participants. A repeated-measures ANOVA with *model* (*text+aesthetic* vs. *text+aesthetic+audio*) and *step* as within-subject factors showed a significant main effect of model, $F(1, 23) = 115.29$, $p < .001$, a significant main effect of step, $F(3, 69) = 40.04$, $p < .001$, and a significant model-by-step interaction, $F(3, 69) = 5.32$, $p = .0023$. A paired-samples t-test on participant-level mean ratings aggregated across steps 40–160 further showed that the text+aesthetic+audio model was rated higher overall than the text+aesthetic model, $t(23) = 10.74$, $p < .001$. Comparing the shared baseline at step 0 with step 160 separately for each model showed significant improvement for both the text+aesthetic model, $t(23) = 5.99$, $p < .001$, and the text+aesthetic+audio model, $t(23) = 8.49$, $p < .001$.

Because the stimuli were a fixed set of generated clips, the inferential tests should be interpreted as participant-level evidence for the sampled stimuli rather than as a full generalization over all possible generations from each model. These results indicate that both reward configurations improve perceived style match but the configuration including audio-reference reward produced stronger gains.

4.4 Diversity collapse

Although reward optimization improved stylistic alignment in the focused experiments, it could also substantially narrow the distribution of generated outputs. In the Chopin experiment, reward values continued to increase over training while the generated samples became increasingly similar, as illustrated by the examples in Play ▶. This behavior is consistent with the diversity collapse previously reported in SMART (Jonason et al., 2025).

The Chopin experiment used group size $G = 16$, KL coefficient $\beta = 0.0015$, learning rate 3×10^{-6}, and reward weights 0.5, 1.0, 0.08, and 0.08 for text alignment, audio alignment, coherence, and musicality, respectively. To quantify changes in diversity, we compute the between-sample Jaccard distance at each training step. Each generated MIDI sample is represented as a set of symbolic musical events encoding timing, pitch, duration, and instrument family. For two samples with event sets A and B, the Jaccard distance is

$$d_J(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}. \quad (1)$$

A value close to 1 indicates that two samples share few events, while a value close to 0 indicates that they are highly similar. For each training step, we report the mean pairwise Jaccard distance across all generated samples from that step.

Table 2: SongEval scores before and after LoRA fine-tuning. Scores are reported on the original SongEval scale.

Dimension	Before fine-tuning	After LoRA fine-tuning	Difference
Coherence	2.665	2.826	0.161
Musicality	2.556	2.667	0.111
Memorability	2.442	2.648	0.206
Clarity	2.383	2.550	0.166
Naturalness	2.394	2.479	0.085
Overall	2.488	2.634	0.146

Table 3: Main fine-tuning settings for the two model variants evaluated in the piano-blues listening test.

Setting	Text + aesthetic	Text + aesthetic + audio
Learning rate	3×10^{-6}	3×10^{-6}
KL coefficient β	0.01	0.01
Text-alignment weight	1.00	1.00
Coherence weight	0.08	0.08
Musicality weight	0.08	0.08
Audio-alignment weight	0.00	1.00

As shown in Figure 4, the mean pairwise Jaccard distance begins near 1.0, indicating high between-sample diversity, but decreases substantially during later training and eventually falls below 0.2.

We also tested whether this behavior could be mitigated using the intra-group similarity penalty described in Appendix A.2. In techno-style tuning, we compared three settings: no similarity penalty with $\beta = 0$, a similarity-penalty weight of 0.1 with $\beta = 0$, and no similarity penalty with stronger KL regularization of $\beta = 0.015$. All runs used group size $G = 20$, learning rate 4×10^{-6}, and audio-alignment weight 1.0.

Without KL regularization or a similarity penalty, reward increased rapidly, but the model again converged toward low-diversity outputs. Adding a strong similarity penalty did not provide a reliable solution: optimization became unstable and began producing invalid MIDI-decoding tokens at approximately step 100. Stronger KL regularization constrained policy movement and better preserved the behavior of the reference model, but it did not give a good balance for the underlying trade-off. Increasing regularization can preserve diversity and stability, but can also make style adaptation difficult.

4.5 Generalization limits under broader prompt distributions

In contrast to previous experiments, we also tested whether the observed improvements extend beyond focused or stylistically related prompt sets. For text-alignment+aesthetic training, we used a more diverse subset of LLM-generated minimal prompts. In our implementation, these statistics are estimated by sampling prompts from SongDescriber (Manco et al., 2023), generating outputs with the frozen base MIDI-LLM model, rendering them with the same SoundFont, and computing each reward component on the resulting audio. Across the broader prompt settings we tested, training was substantially less stable than in the related-style setting.

To isolate the effect of prompt diversity, we compare two experiments using the same training settings and reward weights as in Section 4.2; the only difference is the prompt set used during training. The narrower setting uses related jazz, soul, and lo-fi prompts, while the broader setting uses a more diverse prompt distribution.

As shown in Figure 5, the narrower related-style setting trains comparatively smoothly: reward increases gradually, KL divergence remains moderate, and the gradient norm stays within a stable range. In the broader setting, however, training becomes unstable around step ~ 400. The reward collapses to the invalid-sample fallback value of R_{fail}, while KL divergence and gradient norms spike.

This collapse is associated with failures in the token-to-MIDI conversion stage, as shown in Figure 6. These failures indicate that the model is not merely producing lower-quality music, but increasingly generating malformed symbolic sequences that cannot be decoded into valid MIDI files.

5 DISCUSSION AND LIMITATIONS

In the focused case studies, the model could be pushed toward outputs that were more relevant to the prompt and more consistent with the target style than those produced by the base MIDI-LLM model. This was most clearly supported by the piano-blues listening test, where later fine-tuning checkpoints were judged as stronger matches to the reference style, and where the best-performing condition was the model trained with the full text, aesthetic, and audio-reference reward combination.

The text-audio alignment reward improved broad prompt relevance, but when used alone it could emphasize local stylistic cues at the expense of musical coherence. Adding aesthetic rewards helped stabilize this behavior by encouraging more coherent and musical outputs. The audio-reference reward provided an additional steering signal when suitable reference material was available, and was especially useful in the piano-blues listening test. These results suggest that the strongest behavior comes from combining complementary reward signals rather than relying on a single proxy objective.

Another important result is the trade-off between reward optimization and diversity. In the Chopin experiment, re-

Table 4: Rating summary by condition and training step for retained submissions. Step 0 corresponds to the shared MIDI-LLM baseline before reinforcement-learning fine-tuning and is therefore shown only once.

Condition	Step	Mean	Median	SD
Shared baseline				
MIDI-LLM before RL fine-tuning	0	3.63	3.0	1.97
text+aesthetic				
	40	3.31	3.0	2.08
	80	3.50	3.0	2.42
	120	5.39	5.5	2.59
	160	6.00	7.0	2.23
text+aesthetic+audio				
	40	4.04	4.0	2.21
	80	5.79	6.0	2.66
	120	6.50	7.0	1.92
	160	7.15	7.0	2.11

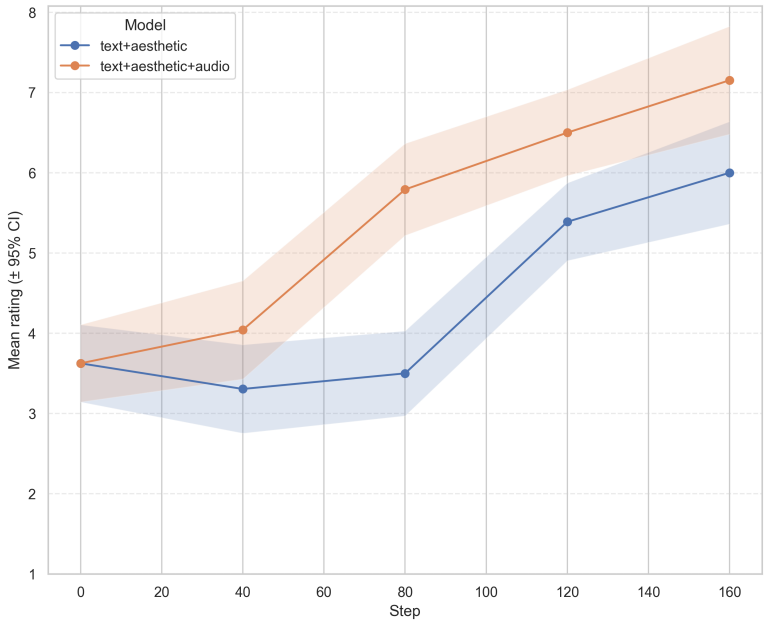

Figure 3: Mean listener rating by training step for the text+aesthetic and text+aesthetic+audio models. Shaded bands indicate 95% confidence intervals around the mean, computed across participant-level condition means.

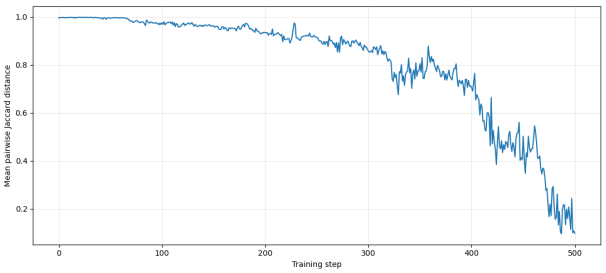

Figure 4: Mean between-sample Jaccard distance over training steps when tuning on the Chopin dataset. It drops sharply in later steps, indicating diversity collapse.

ward continued to increase while the range of generated outputs narrowed sharply. This suggests that style-directed optimization may make the generated outputs more stylistically consistent, but less varied. The pattern is consistent with the diversity collapse reported in SMART, but here it also appears with composite, style-directed audio rewards.

Preserving diversity without weakening adaptation was difficult in our experiments. The intra-group similarity penalty did not provide a reliable solution. Stronger KL regularization better preserved the behavior of the reference policy, but also made adaptation more conservative. The result therefore highlights a broader trade-off between reward improvement, stylistic specialization, output diversity, and policy stability.

The broader-prompt experiments reveal a limitation. Fine-tuning on related jazz, soul, and lo-fi prompts improved held-out text-audio alignment and SongEval scores, suggesting that the method can extend beyond a single prompt or reference example. However, training became substantially less reliable when the prompt distribution was broader. In those settings, optimization could become unstable, KL divergence and gradient norms could spike, and generations could collapse into invalid symbolic outputs.

This instability is partly tied to the symbolic nature of the generator. The reward is computed only after a complete token sequence has been decoded and rendered to audio, but the policy itself acts in a structured MIDI-token space.

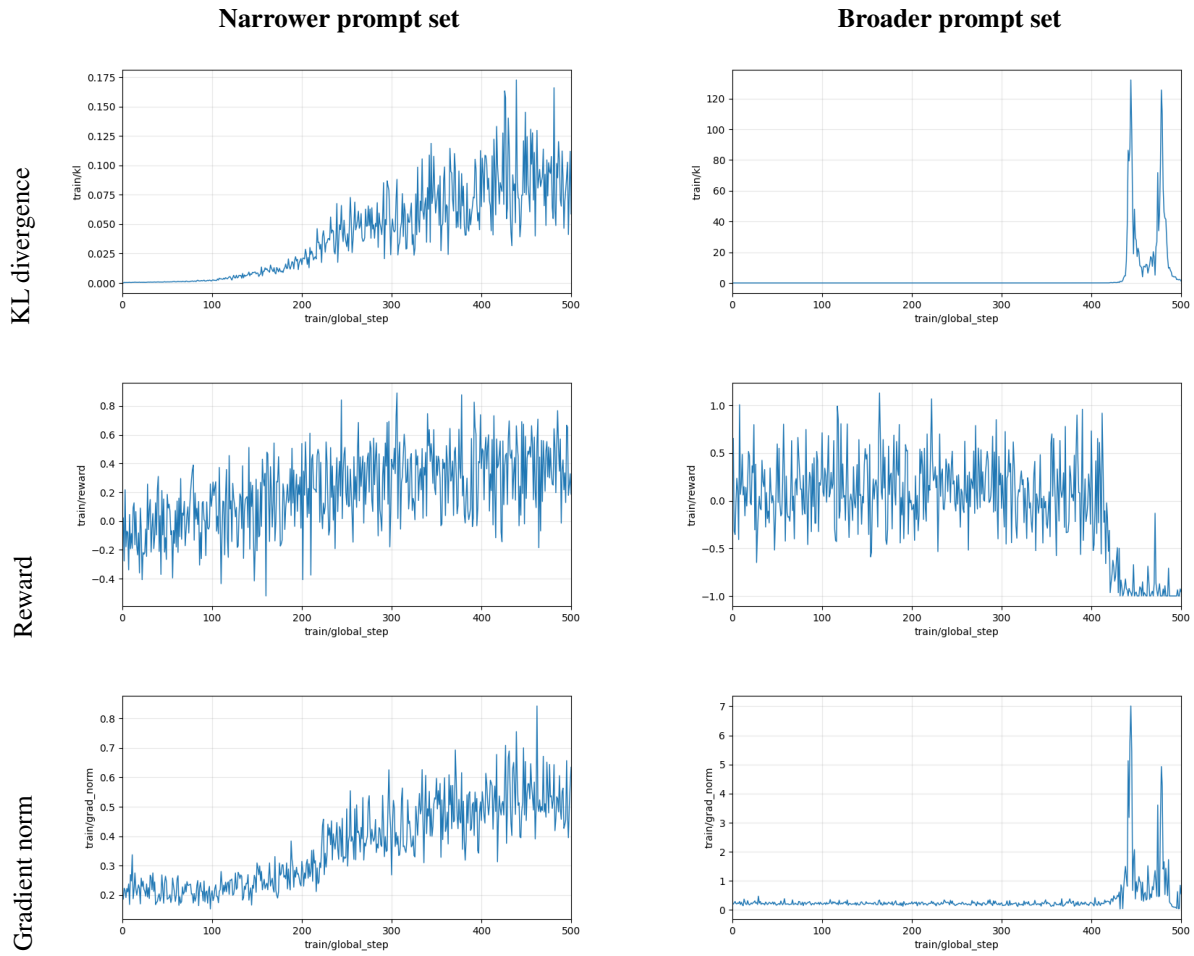

Figure 5: Comparison of training dynamics on narrower and broader prompt sets. The narrower prompt set shows steadily improving reward with comparatively moderate KL divergence and gradient norms. In contrast, the broader prompt set becomes unstable around step ~ 400: KL divergence spikes, and reward collapses toward the invalid-sample fallback value R_{fail}.

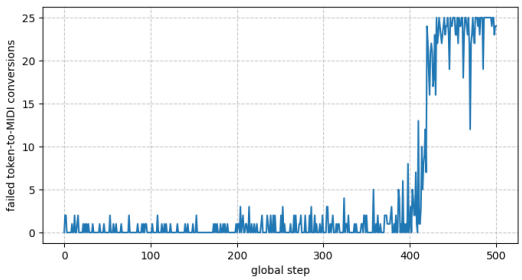

Figure 6: Number of token-to-MIDI conversion failures during broad fine-tuning. The group size is $G = 25$.

When optimization pushes the policy too far from the reference model, generated sequences may no longer satisfy the structural assumptions needed for event-to-MIDI conversion. In these cases, failure is not merely a matter of poor musical quality: the model produces malformed symbolic sequences that cannot be decoded into valid MIDI.

Another limitation is that the reward models are only proxy evaluators. text-audio similarity, aesthetic prediction, and audio-reference similarity do not directly measure musical quality or stylistic correctness. Their behavior depends on the embedding space, prompt wording, reference material, segmentation, layer choice, and rendering pipeline. A generated sample may receive a high reward by matching superficial cues in the embedding space without satisfying the deeper musical properties that a listener would associate with the target style. Conversely, a musically appropriate sample may receive only a moderate score if the embedding model does not represent that style or prompt well. This is why reward improvements should not be interpreted as direct evidence of musical improvement without additional evaluation.

The reward-model issue is especially important because the policy acts in symbolic space while the reward is computed in rendered audio space. The final score can therefore depend not only on the generated notes, but also on the synthesizer, SoundFont, and other audio-side details. This creates a domain mismatch: optimization may exploit properties of the rendering or embedding pipeline rather than improving the symbolic composition itself. Reference-based rewards also inherit this limitation, since they may encourage similarity to particular recordings or production characteristics rather than to a broader musical style.

Finally, the listening test was limited to a single target style, piano blues, and measured perceived style match rather than overall musical quality, originality, long-term structure, or diversity. The results therefore provide useful evidence that the full reward combination can improve perceived stylistic match in one focused setting, but they

should not be taken as evidence that the same configuration generalizes equally well to all musical styles. Broader human evaluations with multiple styles and separate rating dimensions would be needed to establish more general perceptual benefits.

6 CONCLUSION AND FUTURE WORK

This work studied audio-side reinforcement learning for style adaptation in symbolic music generation. We fine-tuned MIDI-LLM with GRPO using composite rewards computed after rendering generated MIDI to audio, combining text-audio alignment, learned aesthetic estimates, and reference-audio similarity. The results show that this framework can improve targeted style control in focused settings. In the piano-blues listening test, fine-tuned models were judged as closer to the target style, with the strongest results from the configuration including text, aesthetic, and audio-reference rewards. A related-style experiment further showed gains on held-out prompts when the prompt distribution remained stylistically coherent. However, the method should not be viewed as a general improvement to MIDI-LLM. Broader prompt distributions made training less stable, and reward optimization could reduce diversity or produce invalid symbolic outputs. Audio-side RL is therefore most promising as a tool for narrow, reference- or style-directed adaptation, while broader use will require stronger controls for stability, symbolic validity, reward quality, and diversity.

Future work should focus on making audio-side reinforcement learning more stable, structurally aware, and interactive. In the current setup, rewards are computed only after a full symbolic sequence has been decoded, which makes the signal sparse and allows the policy to drift into malformed sequences before being penalized. Denser symbolic feedback could reduce invalid-output collapse. Another direction is adaptive reward optimization, since alignment, musical quality, diversity, and symbolic validity can compete with each other under manually chosen scalar weights. Future work should also examine how style-specific fine-tuning affects out-of-domain styles and the model's broader generation abilities, including the trade-off between stronger specialization and reduced generality. Finally, the framework could support interactive style adaptation and HCI studies, where users steer a symbolic generator with prompts, reference tracks, selected examples, or preference feedback.

ETHICS STATEMENT

Our listening test involved participants found through convenience sampling of our institutions. Participants were informed of the task and data collection, and agreed to the conditions before completing the test. No personal identifying information was collected and analyzed. Audio examples were generated by the authors or drawn from royalty-free or public-domain sources. The datasets used to train the base model we use, MIDI-LLM (Wu et al., 2025a), are given in that publication. We use it here to demonstrate the methodology we propose, thus avoiding spending energy to train a new foundational model.

REFERENCES

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. (2016). Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*.

Bhandari, K., Roy, A., Wang, K., Puri, G., Colton, S., and Herremans, D. (2025). Text2midi: Generating symbolic music from captions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 23478–23486.

Chen, H., Smith, J. B. L., Spijkervet, J., Wang, J.-C., Zou, P., Li, B., Kong, Q., and Du, X. (2024). Sympac: Scalable symbolic music generation with prompts and constraints. *arXiv preprint arXiv:2409.03055*.

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. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 8968–8984. PMLR.

Elizalde, B., Deshmukh, S., Al Ismail, M., and Wang, H. (2023). CLAP: Learning audio concepts from natural language supervision. In *ICASSP 2023 – 2023 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1–5. IEEE.

Gao, L., Schulman, J., and Hilton, J. (2023). Scaling laws for reward model overoptimization. In *International Conference on Machine Learning (ICML)*.

Huang, C.-Z. A., Vaswani, A., Uszkoreit, J., Simon, I., Hawthorne, C., Shazeer, N., Dai, A. M., Hoffman, M. D., Dinculescu, M., and Eck, D. (2019). Music transformer: Generating music with long-term structure. In *International Conference on Learning Representations*.

Huang, Q., Jansen, A., Lee, J., Ganti, R., Li, J. Y., and Ellis, D. P. W. (2022). MuLan: A joint embedding of music audio and natural language. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference*, pages 559–566.

Huang, Y.-S. and Yang, Y.-H. (2020). Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1180–1188.

Jonason, N., Casini, L., and Sturm, B. L. T. (2025). SMART: Tuning a symbolic music generation system with an audio domain aesthetic reward. In *Proceedings of the 6th Conference on AI Music Creativity*, Brussels, Belgium.

Kirk, R., Mediratta, I., Nalmpantis, C., Luketina, J., Hambro, E., Grefenstette, E., and Raileanu, R. (2024). Understanding the effects of RLHF on LLM generalisation and diversity. In *The Twelfth International Conference on Learning Representations*.

Li, Y., Yuan, R., Zhang, G., Ma, Y., Chen, X., Yin, H., Xiao, C., Lin, C., Ragni, A., Benetos, E., Gyenge, N., Dannenberg, R., Liu, R., Chen, W., Xia, G., Shi, Y., Huang, W., Wang, Z., Guo, Y., and Fu, J. (2024). MERT: Acoustic music understanding model with large-scale self-supervised training. In *International Conference on Learning Representations*.

Manco, I., Weck, B., Doh, S., Won, M., Zhang, Y., Bogdanov, D., Wu, Y., Chen, K., Tovstogan, P., Benetos, E., Quinton, E., Fazekas, G., and Nam, J. (2023). The

song describer dataset: a corpus of audio captions for music-and-language evaluation. In *Proceedings of the NeurIPS Workshop on Machine Learning for Audio*.

Manheim, D. and Garrabrant, S. (2018). Categorizing variants of goodhart's law. *arXiv preprint arXiv:1803.04585*.

Melechovsky, J., Roy, A., and Herremans, D. (2024). Midicaps: A large-scale midi dataset with text captions. *arXiv preprint arXiv:2406.02255*.

Meta (2024). meta-llama/llama-3.2-1b (model card). https://huggingface.co/meta-llama/Llama-3.2-1B. Hugging Face; release date Sept 25, 2024.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y., et al. (2024). Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Thickstun, J., Hall, D. L. W., Donahue, C., and Liang, P. (2024). Anticipatory music transformer. *Transactions on Machine Learning Research*.

Tjandra, A., Wu, Y.-C., Guo, B., Hoffman, J., Ellis, B., Vyas, A., Shi, B., Chen, S., Le, M., Zacharov, N., et al. (2025). Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*.

Wang, Y., Wu, S., Hu, J., Du, X., Peng, Y., Huang, Y., Fan, S., Li, X., Yu, F., and Sun, M. (2025). NotaGen: Advancing musicality in symbolic music generation with large language model training paradigms. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 10207–10215.

Wu, S.-L., Kim, Y., and Huang, C.-Z. A. (2025a). MIDI-LLM: Adapting large language models for text-to-MIDI music generation. In *Proceedings of the NeurIPS 2025 Workshop on AI for Music*.

Wu, Y., Brade, S., Ma, A. T., Fowler, T.-J., Yang, E., Banar, B., Courville, A., Jaques, N., and Huang, C.-Z. A. (2025b). Generative adversarial post-training mitigates reward hacking in live human-ai music interaction. *arXiv preprint arXiv:2511.17879*.

Xu, W., McAuley, J., Berg-Kirkpatrick, T., Dubnov, S., and Dong, H.-W. (2025). Generating symbolic music from natural language prompts using an LLM-enhanced dataset. In *Proceedings of the 26th International Society for Music Information Retrieval Conference*, pages 215–222.

Yao, J., Ma, G., Xue, H., Chen, H., Hao, C., Jiang, Y., Liu, H., Yuan, R., Xu, J., Xue, W., et al. (2025). Songeval: A benchmark dataset for song aesthetics evaluation. *arXiv preprint arXiv:2505.10793*.

Zhu, H., Zhou, Y., Chen, H., Yu, J., Ma, Z., Gu, R., Luo, Y., Tan, W., and Chen, X. (2025). Muq: Self-supervised music representation learning with mel residual vector quantization.

A APPENDIX

AI-use disclosure We used OpenAI GPT 5.4 to generate candidate text prompts for the jazz/soul/lo-fi prompt set. The model was instructed to produce prompts describing jazz and closely related genres, without vocals, and specify relevant instrumentation and mood separated by commas. The generated prompts were manually inspected before use.

A.1 Group Relative Policy Optimization objective

For each prompt q, GRPO samples a group of G completions

$$\{o_i\}_{i=1}^{G} \sim \pi_{\theta_{\text{old}}}(\cdot \mid q),$$

and assigns each completion a scalar reward $r_i = r(q, o_i)$. The rewards are normalized within the group to obtain a completion-level advantage:

$$\hat{A}_i = \frac{r_i - \mu_r}{\sigma_r + \epsilon}, \qquad \mu_r = \frac{1}{G}\sum_{i=1}^{G} r_i, \qquad \sigma_r = \sqrt{\frac{1}{G}\sum_{i=1}^{G}(r_i - \mu_r)^2}.$$

Thus, each completion is judged relative to the other completions sampled for the same prompt.

Let

$$\rho_{i,t}(\theta) = \frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} \mid q, o_{i,<t})}$$

be the token-level probability ratio between the updated policy and the sampling policy. GRPO uses a PPO-style clipped objective with a KL penalty to a fixed reference policy π_{ref}:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|}\left[\min\Big(\rho_{i,t}(\theta)\hat{A}_i,\ \text{clip}(\rho_{i,t}(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_i\Big) - \beta\, D_{\text{KL}}(\pi_\theta(\cdot \mid q, o_{i,<t}) \,\|\, \pi_{\text{ref}}(\cdot \mid q, o_{i,<t}))\right].$$

The policy is updated to maximize $\mathcal{J}_{\text{GRPO}}$. The clipping term limits the size of each policy update, while the KL term keeps the policy close to the frozen reference model. In our setting, the reward r_i is computed only after the generated symbolic sequence has been decoded to MIDI and rendered to audio.

A.2 Intra-group similarity penalty

Let $\psi(\cdot)$ be an audio embedding function that maps each generated audio sample x_i in a group to a single embedding vector. For a group of G generated samples, define the pairwise similarity between two samples as

$$\text{sim}_{i,j} = \cos(\psi(x_i), \psi(x_j)), \qquad i \neq j.$$

The similarity penalty for sample x_i is

$$r_{\text{div}}(x_i) = -\frac{1}{G-1}\sum_{\substack{j=1 \\ j\neq i}}^{G} \text{sim}_{i,j}.$$

Thus, samples that are highly similar to the rest of the group receive a lower reward. In our implementation, $\psi(\cdot)$ can be instantiated using MuQ embeddings, similarly to the audio–audio alignment reward.

Table 5: Default hyperparameters used for GRPO fine-tuning unless specified otherwise in the experiments. Reward weights are applied only when the corresponding reward component is used; otherwise, the respective weight is set to zero.

Hyperparameter	Value
Base model	MIDI-LLM (Llama 3.2 1B)
Precision	bfloat16
Quantization	none
Learning rate	4×10^{-6}
KL coefficient β	0.001
Scheduler	cosine
Gradient clipping	1.0
Gradient accumulation	1
Generations per prompt G	25
Per-device batch size	5
Temperature	1.0
Top-p	0.97
Max prompt length	32
Max completion length	1000
Sample rate	24 kHz
Max audio duration	70 s
SoundFont	FluidR3_GM.sf2
Reward weights	coherence 0.08 musicality 0.08 text-audio alignment 1.0 audio–audio alignment 1.0