MIDIFILL: TEXT-CONDITIONED SYMBOLIC MUSIC INFILLING

Jeonghyeon Im[1,*] **Shuyu Li**[2,*] **Dooho Choi**[1,*]
Yunsick Sung[1,†]
[1] Department of Computer Science and Artificial Intelligence, Dongguk University-Seoul, Seoul, 04620, Republic of Korea
[2]NUI/NUX Platform Research Center, Dongguk University, Republic of Korea
jounghyunjet@dgu.ac.kr, lishuyu@dongguk.edu, likeb789@dgu.ac.kr, sung@dongguk.edu
[*] These authors contributed equally. [†] Corresponding author.

ABSTRACT

Symbolic music infilling, the task of completing missing bars considering the corresponding musical contexts, remains underexplored in a text-conditioned setting. We present MidiFill, a Discrete Diffusion Transformer (DDiT) that reconstructs masked bars of MIDI sequences guided by natural-language captions, using a frozen BERT encoder for text conditioning and Classifier-Free Guidance (CFG) at inference time. Experiments across 2-bar, 4-bar, and 8-bar masking conditions demonstrate that MidiFill achieves CLAP scores within 0.003 of ground-truth (GT) recordings and surpasses GT on approximately 48–50% of samples under CLAP-based comparisons. These results suggest that text-conditioned infilling can produce musically plausible and semantically competitive completions even for large masked spans, establishing a strong baseline for the symbolic music infilling task.

1. INTRODUCTION

Symbolic music generation has undergone rapid progress with the introduction of transformer-based sequence models and, more recently, diffusion-based generative frameworks [1]. These systems operate over discrete token representations of MIDI, enabling fine-grained control over pitch, rhythm, and dynamics at the note level. Unlike audio-domain generation, symbolic generation preserves full editability of the musical content, making it highly relevant for computer-assisted composition, music production, and interactive creativity tools. Controllable symbolic generation, where generation is guided by user-specified attributes such as genre, mood, or instrumentation, has therefore attracted growing research interest [2, 3].

Among the various generation scenarios, music infilling, the task of reconstructing a missing segment within an existing musical context given both a preceding and a following passage, remains relatively underexplored in the symbolic domain. Infilling is practically important: in real-world workflows, composers routinely sketch pieces with intentional gaps and expect a generative system to propose contextually coherent completions. For instance, a composer may have a strong opening theme and a satisfying resolution, but require assistance bridging them; a producer may wish to replace a repetitive loop segment with something more dynamic while preserving the surrounding arrangement. Filling such gaps demands bidirectional understanding of musical context, a challenge that purely autoregressive, left-to-right models are structurally ill-suited to address.

Existing symbolic infilling approaches, such as masked language model-based method [4], provide bidirectional context awareness but lack semantic conditioning. These models generate completions that are locally consistent with the surrounding notes, but offer no mechanism for the user to specify the intended mood, style, or character of the inserted passage. Separately, text-conditioned audio generation systems [5] have demonstrated that natural language is a powerful and intuitive conditioning modality for music; however, direct analogues in the symbolic domain are absent. As a result, no existing system supports the practically important setting of text-guided symbolic music infilling, where the generated passage must be simultaneously coherent with its musical context and semantically aligned with a natural-language description.

To address this gap, we propose MidiFill, a text-conditioned MIDI infilling model that integrates discrete diffusion with cross-modal semantic conditioning. MidiFill operates on a REMI token representation and takes as input an incomplete MIDI sequence with a contiguous masked span, together with a natural-language caption (e.g., "upbeat jazz with syncopated rhythm"). Text captions are encoded using a frozen BERT encoder [6], and the resulting sequence of contextual embeddings is injected into every DDiT [7] layer via cross-attention. Classifier-Free Guidance (CFG) [8] is applied at inference time to amplify the text condition, and the model iteratively denoises the masked token span over multiple steps conditioned jointly on the surrounding MIDI context and the text embedding.

The main contributions of this work are as follows:

1. We propose MidiFill, a discrete diffusion model for symbolic music infilling with cross-modal text con-

© 2026 Korean Society for Music Informatics. **Citation:** J. Im, S. Li, D. Choi, and Y. Sung, "MidiFill: Text-Conditioned Symbolic Music Infilling", *The First Korean Society for Music Informatics Conference*, Seoul, Republic of Korea, 2026.

ditioning via a frozen BERT encoder, enabling semantically steerable completion of missing MIDI passages.

2. We show that MidiFill generalizes beyond its training distribution, maintaining stable performance on 8-bar evaluation conditions despite being trained only with 1-4 bar masks.

3. We demonstrate through automatic CLAP-based evaluation that MidiFill generates completions with near ground-truth semantic fidelity ($|\Delta\text{CLAP}| \leq 0.003$), achieving win rates of 48–50% against GT even for large 8-bar masked spans.

2. RELATED WORK

Symbolic music generation has been approached through autoregressive transformers [1–3, 9–11] and, more recently, text-conditioned and diffusion-based frameworks operating over discrete MIDI token sequences. In the symbolic infilling setting, masked language model-based methods such as the Masked Music Model (MMM) [4] and BERT-pretrained models including MusicBERT [12] and MidiBERT-Piano [13] have demonstrated that bidirectional attention over MIDI tokens enables effective bar-level and note-level reconstruction from surrounding context. However, none of these works incorporate text conditioning, leaving the generated content guided solely by the musical context without semantic control. In the audio domain, diffusion models have achieved strong results in text-conditioned generation [5, 14], and discrete masked diffusion [7] has extended this paradigm to categorical token sequences via an absorbing state, which is particularly well-suited to symbolic music due to its compatibility with token-level MIDI representations. MidiFill builds on this formulation by integrating cross-modal BERT-based text conditioning into the discrete diffusion framework, enabling semantically steerable infilling in the symbolic domain.

3. METHOD

3.1 MIDI Representation and Masking

We represent MIDI files using the REMI tokenization scheme [15] as implemented in MidiTok [16]. Each musical event is encoded as a sequence of tokens drawn from a vocabulary covering Pitch, Duration, Position, Program, Tempo, TimeSignature, and Bar events, with special tokens `[PAD]`, `[BOS]`, `[EOS]`, and `[MASK]`. The vocabulary size is approximately 500–1000 tokens depending on the granularity settings, and a complete piece is represented as a flat token sequence $\mathbf{x} = (x_1, x_2, \ldots, x_T)$ with a maximum sequence length of $T = 1024$ tokens.

For infilling, we apply a contiguous bar-level masking strategy: bar boundaries are identified via `Bar_None` tokens, and a random contiguous span of $m \in \{1, 2, 3, 4\}$ bars is selected during training. All tokens within the selected span are replaced with the `[MASK]` token, while the left context $\mathbf{x}_{1:b-1}$ and right context $\mathbf{x}_{b+m:T}$ are retained and provided to the model bidirectionally. This contiguous masking is more challenging than random token-level masking, as the model must reconstruct coherent melodic and rhythmic structures spanning entire bars from the surrounding context and the text signal alone. At evaluation time, we assess performance on masking spans of $\{2, 4, 8\}$ bars to probe a wider range of task difficulties.

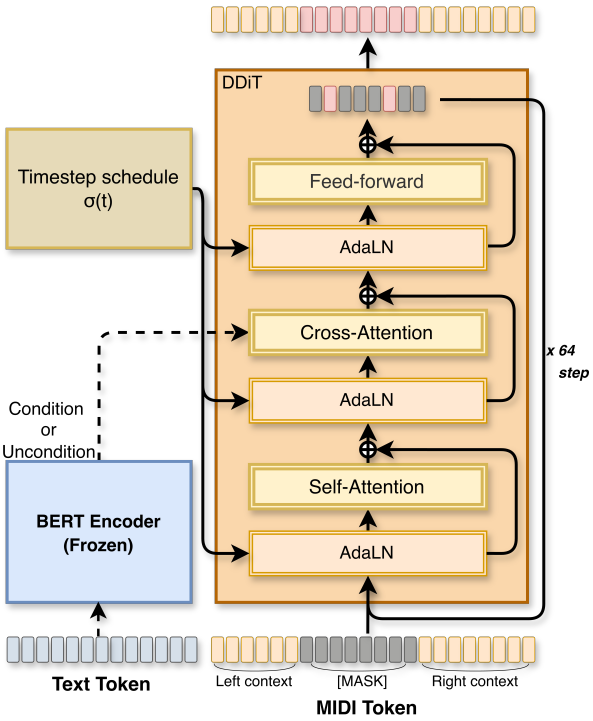

Figure 1. Overview of MidiFill. A contiguous bar span is masked in the REMI token sequence. The frozen BERT encoder produces text embeddings that are injected via cross-attention at every DDiT block. The timestep σ conditions each block through AdaLN. The model iteratively denoises the masked span over multiple steps, guided by both the surrounding MIDI context and the text signal.

3.2 Model Architecture

The overall architecture of MidiFill is illustrated in Figure 1. MidiFill adopts a discrete absorbing-state diffusion formulation [7] with the *subs* parameterization. The forward process defines a masking distribution over infill positions: at noise level σ, each infill token is masked with probability $1 - e^{-\sigma}$ and revealed as the ground-truth token otherwise, with context tokens remaining unchanged throughout. The noise level follows a LogLinear schedule:

$$\sigma(t) = -\log\big(1 - (1 - \varepsilon)\, t\big), \quad t \in [0, 1], \tag{1}$$

with $\varepsilon = 10^{-3}$, yielding $\sigma_{\min} \approx 0.001$ and $\sigma_{\max} \approx 6.9$. The reverse process is parameterized by a Discrete Diffusion Transformer (DDiT) with $L = 12$ transformer blocks, hidden dimension $d = 768$, and 12 attention heads (head dimension 64), totaling approximately

130M trainable parameters (excluding the frozen BERT encoder). Each block consists of (1) self-attention with FlashAttention-2 [17] over the full MIDI token sequence, (2) cross-attention to the projected BERT text embeddings, (3) a feed-forward MLP with expansion ratio 4 and GELU activation and (4) Adaptive Layer Normalization (AdaLN) modulated by the diffusion timestep embedding. For text conditioning, a frozen `bert-base-uncased` encoder [6] maps a caption $\mathbf{t}$ to contextual embeddings $\mathbf{C} = \text{BERT}(\mathbf{t}) \in \mathbb{R}^{128\times768}$, which are projected via a learned linear layer and injected into every DDiT block through cross-attention, allowing each layer to attend over the non-padding text token representations. The timestep σ is encoded via a sinusoidal TimestepEmbedder with a two-layer MLP (Linear $\rightarrow$ SiLU $\rightarrow$ Linear), followed by an additional SiLU activation, yielding a conditioning vector of dimension 256; the AdaLN modulation applies per-block scale and shift to both the attention and MLP sub-layers. Rotary Position Embeddings (RoPE) [18] are applied to the query and key projections in self-attention; the value projection passes through the same operation with an identity rotation, leaving it unchanged. The training objective is:

$$\mathcal{L} = -\mathbb{E}\left[\sum_{i\in\mathcal{M}} \log p_\theta(x_i \mid \mathbf{x}_{\setminus\mathcal{M}}, \mathbf{C}) \cdot \frac{\mathrm{d}\sigma}{e^\sigma - 1}\right], \quad (2)$$

where $\mathcal{M}$ denotes the masked positions and the weighting term follows the subs parameterization. Logits at unmasked positions are set to zero to preserve the original context tokens.

3.3 Training Strategy and Inference

To enable Classifier-Free Guidance (CFG) [8] at inference time, text conditioning is dropped with probability 0.1 during training by replacing the caption with an empty string, allowing the model to learn both conditional and unconditional score estimates within a single training run. To reduce variance in gradient estimation, antithetic timestep sampling is applied across the batch, and an exponential moving average (EMA) of model weights with decay $= 0.9999$ is maintained throughout training and used exclusively at inference time. At inference, the reverse diffusion loop runs for 64 steps over timesteps linearly spaced from $t = 1$ to $t = 10^{-5}$. CFG is applied in logit space at each step:

$$\tilde{\ell}(x_i) = \ell_{\text{uncond}}(x_i) + \gamma\left(\ell_{\text{cond}}(x_i) - \ell_{\text{uncond}}(x_i)\right), \quad (3)$$

where ℓ denotes pre-softmax logits and $\gamma = 2.0$ is the CFG scale. Tokens are sampled from the resulting categorical distribution and written back only to positions that are both still masked and within the infilling region.

4. EXPERIMENTS

4.1 Dataset and Training

We trained MidiFill on MidiCaps [19], a dataset of MIDI files paired with natural-language captions. The dataset was split by the `test_set` field: samples with `test_set=False` were used for training and those with `test_set=True` for validation. Each training sample was constructed by tokenizing a MIDI file with the REMI scheme, randomly masking a contiguous span of 1–4 bars, and pairing the result with its corresponding text caption.

The model was trained for 200,000 steps using the AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay $= 0.01$, gradient clip $= 1.0$) with a peak learning rate of 3×10^{-4} and a linear warmup of 1,000 steps. Training used a total batch size of 48 (24 per H100 GPU $\times$ 2 GPUs) with bfloat16 precision, distributed via HuggingFace Accelerate with DeepSpeed ZeRO-2.

4.2 Evaluation Metrics

We evaluated MidiFill on a held-out test set using $N = 50$ samples per masking condition. Since MIDI cannot be directly evaluated using audio-domain metrics without rendering, we synthesized all MIDI outputs to WAV at 44.1 kHz using `pretty_midi` and `pyfluidsynth`. We evaluate across three masking conditions (2-bar, 4-bar, and 8-bar spans) and report the following metrics:

CLAP Score. The cosine similarity between the CLAP [20] audio embedding of the rendered output and the CLAP text embedding of the corresponding caption, measuring semantic alignment with the text description. CLAP scores are computed using the HTSAT-base model pretrained on AudioSet via the `laion_clap` library, with audio processed in 10-second chunks with a 5-second hop.

CLaMP3 Score. The cosine similarity between CLaMP3 [21] symbolic music embeddings of the infilled output and the GT recording, measuring music-domain semantic similarity independent of audio rendering quality.

Win Rate. The fraction of test samples for which the infilled output achieved a higher CLAP or CLaMP3 score than the corresponding GT recording rendered under identical conditions. A win rate near 50% indicates that the infilled output is semantically competitive with the original recording.

4.3 Results

Table 1 presents the full evaluation results. MidiFill consistently achieved Infill CLAP scores closely tracking GT CLAP scores across all masking conditions, with deltas ranging from -0.001 (4-bar) to -0.003 (8-bar), all well within one standard deviation of the GT scores. CLaMP3 scores similarly show small absolute gaps (-0.016 to -0.020), confirming that MidiFill produces semantically consistent outputs across both text-based and music-domain evaluation spaces.

4.4 Analysis

The CLAP delta remained stable across masking conditions, ranging from -0.001 (4-bar) to -0.003 (8-bar), indicating that the semantic gap between infilled output and GT did not grow substantially as the masked span increased from 2 to 8 bars, suggesting that bidirectional at-

Table 1. Evaluation results (50 samples, 64 denoising steps, CFG = 2.0). ↑ = higher is better.

Mask	CLAP ↑			CLaMP3 ↑		
	GT	MidiFill	Win Rate	GT	MidiFill	Win Rate
2 bar	0.331 ± 0.078	0.330 ± 0.095	48%	0.159 ± 0.045	0.143 ± 0.056	38%
4 bar	0.333 ± 0.080	0.332 ± 0.095	48%	0.160 ± 0.046	0.140 ± 0.056	34%
8 bar	0.339 ± 0.073	0.336 ± 0.091	50%	0.158 ± 0.045	0.140 ± 0.056	38%

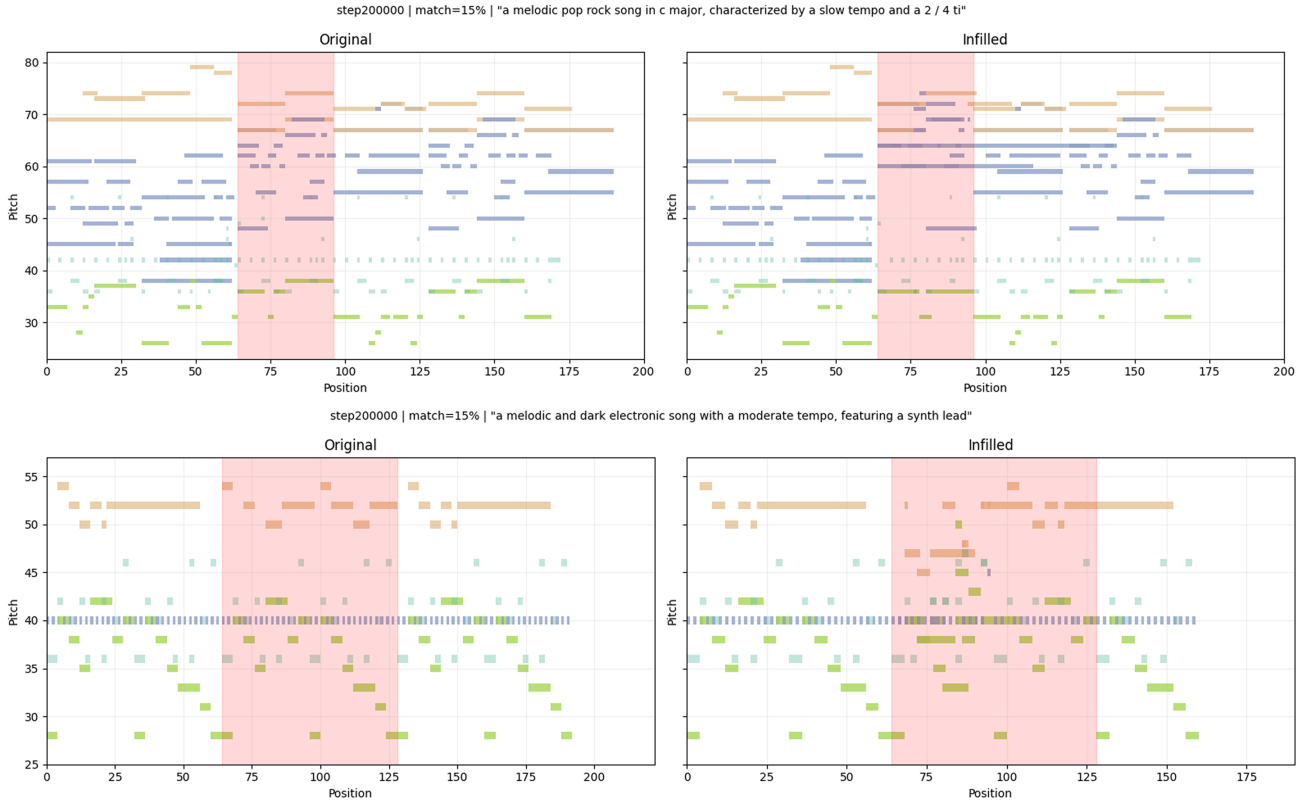

Figure 2. Piano roll visualizations of original and MidiFill-infilled MIDI. Shaded regions indicate the masked span; each color represents a distinct instrument track. Match rate denotes token-level exact match with GT within the masked span. (Top) 1-bar masking. (Bottom) 2-bar masking.

tention in the discrete diffusion backbone effectively leverages both left and right context simultaneously, an advantage over autoregressive models that condition only on past tokens. The CLAP win rates of 48%, 48%, and 50% further suggest that MidiFill's outputs are sometimes more semantically aligned with the text caption than the original recording, which may reflect that GT recordings are not always optimally aligned with high-level textual descriptions derived from metadata. Interestingly, both Infill and GT CLAP scores increased slightly with longer masking spans ($0.330 \rightarrow 0.336$ for infill; $0.331 \rightarrow 0.339$ for GT), which we hypothesize reflects a selection effect: test samples with longer masked spans tended to feature more harmonically rich and expressive regions more distinctively described by their captions.

4.5 Case Study

Figure 2 presents two representative infilling examples visualized as piano rolls. In both cases, MidiFill generates completions that maintain the pitch register and rhythmic density of the surrounding context, despite low token-level match rates (15%). This confirms that exact token match is not a reliable proxy for musical quality: the model produces plausible alternatives that differ from GT at the token level but remain coherent within the musical and textual context. In the first example (top), a 1-bar masked span in a melodic pop rock piece is filled with notes that preserve the harmonic structure and melodic contour of the adjacent bars. In the second example (bottom), a 2-bar masked span in a sparse electronic piece is reconstructed with rhythmic patterns and pitch choices consistent with the synth-dominant texture described in the caption.

5. CONCLUSION

We presented MidiFill, a text-conditioned MIDI infilling model that combines discrete absorbing-state diffusion with BERT-based semantic conditioning. Evaluation

across 2-bar, 4-bar, and 8-bar masking conditions on MidiCaps showed that MidiFill achieves near-GT semantic fidelity under CLAP-based evaluation (delta ≤ 0.003) with win rates of 48–50%. These results establish MidiFill as a competitive baseline for text-guided symbolic music infilling and highlight the potential of cross-modal conditioning for creative music applications.

6. ACKNOWLEDGEMENT

This research was supported by the Strengthening AI Computing Resource Utilization Foundation, funded by the Government of the Republic of Korea (Ministry of Science and ICT)

7. References

[1] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. M. Dai, M. D. Hoffman, and D. Eck, "Music transformer: Generating music with long-term structure," in *International Conference on Learning Representations (ICLR)*, 2019. [Online]. Available: https://arxiv.org/abs/1809.04281

[2] D. von Rütte, L. Biggio, Y. Kilcher, and T. Hofmann, "FIGARO: Generating symbolic music with fine-grained artistic control," in *International Conference on Learning Representations (ICLR)*, 2023. [Online]. Available: https://arxiv.org/abs/2201.10936

[3] P. Lu, X. Xu, C. Kang, B. Yu, C. Xing, X. Tan, and J. Bian, "MuseCoco: Generating symbolic music from text," *arXiv preprint arXiv:2306.00110*, 2023. [Online]. Available: https://arxiv.org/abs/2306.00110

[4] J. Ens and P. Pasquier, "MMM: Exploring conditional multi-track music generation with the transformer," *arXiv preprint arXiv:2008.06048*, 2020. [Online]. Available: https://arxiv.org/abs/2008.06048

[5] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023. [Online]. Available: https://arxiv.org/abs/2306.05284

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT*, 2019, pp. 4171–4186. [Online]. Available: https://arxiv.org/abs/1810.04805

[7] S. S. Sahoo, M. Arriola, A. Gokaslan, E. M. Marroquin, A. M. Rush, Y. Schiff, J. T. Chiu, and V. Kuleshov, "Simple and effective masked diffusion language models," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. [Online]. Available: https://arxiv.org/abs/2406.07524

[8] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022. [Online]. Available: https://arxiv.org/abs/2207.12598

[9] R. Yuan, H. Lin, Y. Wang, Z. Tian, S. Wu, T. Shen, G. Zhang *et al.*, "ChatMusician: Understanding and generating music intrinsically with llm," *arXiv preprint arXiv:2402.16153*, 2024. [Online]. Available: https://arxiv.org/abs/2402.16153

[10] K. Bhandari, A. Roy, K. Wang, G. Puri, S. Colton, and D. Herremans, "Text2midi: Generating symbolic music from captions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 22, 2025, pp. 23 478–23 486. [Online]. Available: https://arxiv.org/abs/2412.16526

[11] S. Li, D. Choi, and Y. Sung, "Midilm: A dual-path model for controllable text-to-midi generation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 28, pp. 23 160–23 168, 2026.

[12] M. Zeng, X. Tan, R. Wang, Z. Ju, T. Qin, and T.-Y. Liu, "MusicBERT: Symbolic music understanding with large-scale pre-training," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 791–800. [Online]. Available: https://arxiv.org/abs/2106.05630

[13] Y.-H. Chou, I.-C. Chen, C.-J. Chang, J. Ching, and Y.-H. Yang, "MidiBERT-Piano: Large-scale pre-training for symbolic music understanding," *arXiv preprint arXiv:2107.05223*, 2021. [Online]. Available: https://arxiv.org/abs/2107.05223

[14] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "AudioLDM: Text-to-audio generation with latent diffusion models," in *International Conference on Machine Learning (ICML)*, 2023. [Online]. Available: https://arxiv.org/abs/2301.12503

[15] Y.-S. Huang and Y.-H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," in *ACM International Conference on Multimedia (MM)*, 2020, pp. 1180–1188. [Online]. Available: https://arxiv.org/abs/2002.00212

[16] N. Fradet, J.-P. Briot, F. Chhel, A. El Fallah Seghrouchni, and N. Gutowski, "MidiTok: A python package for MIDI file tokenization," in *Extended Abstracts for the Late-Breaking Demo Session of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021. [Online]. Available: https://arxiv.org/abs/2310.17202

[17] T. Dao, "FlashAttention-2: Faster attention with better parallelism and work partitioning," in *International Conference on Learning Representations (ICLR)*, 2024. [Online]. Available: https://arxiv.org/abs/2307.08691

[18] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, "RoFormer: Enhanced transformer with rotary position embedding," *arXiv preprint arXiv:2104.09864*, 2021. [Online]. Available: https://arxiv.org/abs/2104.09864

[19] J. Melechovsky, A. Roy, and D. Herremans, "MidiCaps: A large-scale MIDI dataset with text captions," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, 2024. [Online]. Available: https://arxiv.org/abs/2406.02255

[20] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023. [Online]. Available: https://arxiv.org/abs/2211.06687

[21] S. Wu, G. Zhancheng, R. Yuan, J. Jiang, S. Doh, G. Xia, J. Nam, X. Li, F. Yu, and M. Sun, "Clamp 3: Universal music information retrieval across unaligned modalities and unseen languages," in *Findings of the Association for Computational Linguistics: ACL 2025*, 2025, pp. 2605–2625.