PianoGen: Piano Continuation and Generation through Deep Learning

Nathan Yao
Department of Machine Learning
Carnegie Mellon University
Pittsburgh, PA 15213
myao@andrew.cmu.edu

Melody Gao
Department of Machine Learning
Carnegie Mellon University
Pittsburgh, PA 15213
melodyg@andrew.cmu.edu

Abstract

Music generation poses the challenge of balancing short-term coherence with long-term musical structure. PianoGen is a deep learning framework for piano music continuation directly from raw audio. The system employs a SoundStream encoder to convert waveforms into discrete tokens, which a Transformer then models autoregressively to generate continuations. This token-based approach preserves expressive performance details without relying on symbolic data such as MIDI. Using the MAESTRO dataset, PianoGen is trained and evaluated via negative log-likelihood (NLL) to assess the quality and coherence of generated sequences. Preliminary results demonstrate that the model produces smooth and realistic short-term continuations, though long-term structure remains a challenge. Future work includes introducing hierarchical transformer layers and musically informed embeddings to enhance structural consistency and perceptual realism.

1 Introduction

Piano compositions encompass a wide range of features such as mood, difficulty, motifs, harmony, rhythm, form, and expressive techniques that together shape the character and overall listening experience of a piece. Music generation, more specifically music continuation, faces the challenge of maintaining both short-term characteristics, such as pitch and phrasing, and long-term characteristics, such as rhythm, structure, and thematic development. Achieving this balance without producing music that merely replicates or repeats existing patterns remains a significant difficulty in the field.

We chose to explore this problem due to our shared interest in music theory and in understanding the elements of musical composition that make a piece engaging to the listener. In this study, we utilize the Maestro Dataset (Hawthorne et al. 2019), which contains approximately 200 hours of piano performance recordings spanning a range of performance qualities and compositional styles.

To generate continuations of piano performances, we first implement a baseline model in which a Soundstream encoder converts audio into discrete tokens across 16 quantization levels. The primary quantization level is then provided to a transformer, which predicts subsequent tokens, and the predicted tokens are decoded back into audio using the corresponding Soundstream decoder. We train and evaluate this baseline using negative log-likelihood (NLL), which measures the quality and coherence of the generated continuations.

To improve upon this baseline, we retain the Soundstream encoder but introduce a multi-output transformer that conditions on the first four quantization levels and then predicts the next four levels of tokens. As before, the predicted tokens are decoded with the Soundstream decoder. This multi-output model is trained and evaluated using an estimate of log Mel-spectrogram loss computed in the tokenized space, providing a more perceptually grounded metric.

Preprint. Under review.

2 Literature Review

2.1 WaveNet

The WaveNet model (Aaron van den Oord et al. 2016) represented the first major breakthrough in generating realistic musical and speech audio directly from neural networks. WaveNet employed an autoregressive architecture, where each audio sample is generated conditioned on previous samples. While this approach proved highly effective for speech synthesis, it struggled to maintain long-term temporal coherence required for structured music such as piano compositions.

2.2 Autoregressive Discrete Autoencoders

Dieleman (Dieleman, Aäron van den Oord, and Simonyan 2018) investigated the limitations of autoregressive models to capture long-range correlations in music. Additionally, they emphasize the use of raw audio data rather than high level representations such as MIDI, which abstract away certain idiosyncrasies which are important to the human perception of musicality and thus enjoyment. To address the issue of maintaining long term temporal coherence, they trained a hierarchy of Vector Quantized Variation Autoencoders (VQ-VAE) models and used a WaveNet decoder to turn the audio to raw waveforms. This hierarchical structure enabled the model to compress and represent long-term structure while discarding low-level redundancy, achieving coherent long-form music generation with realistic computational demands. This work directly inspires our inclusion of a Soundstream-based audio codec, which extends the principles of the VQ-VAE hierarchy.

2.3 Jukebox

The impact of the VQ-VAE approach led to OpenAI's Jukebox (Dhariwal et al. 2020), which expanded on this concept by introducing multi-scale VQ-VAE compression of raw audio into discrete codes. These codes were then modeled with autoregressive Transformers, introducing hierarchical modeling that differs from Dieleman's single-resolution method. This hierarchical compression allowed Jukebox to generate coherent compositions spanning several minutes, making it one of the first systems to scale neural music generation to full-length songs. By quantizing continuous waveforms into discrete tokens, Jukebox enabled the use of Transformer architectures, which had previously been impractical for high-rate audio.

2.4 AudioLM

Finally, AudioLM (Borsos et al. 2022) extended these ideas beyond music to general audio generation, removing the need for domain-specific training data or text conditioning. AudioLM first compresses raw waveforms into discrete tokens using a Soundstream encoder, then divides these into semantic and acoustic tokens to separately model long-term structure and fine-grained sound detail. The model predicts token sequences using self-supervised learning, and reconstructs audio via the Soundstream decoder. AudioLM introduced self-supervised, text free generation, improving on previous models which relied on textual conditioning. Overall, it improved generative diversity and realism beyond music generation.

3 Methods and Model

3.1 Baseline Architecture

Our goal was to take raw WAV piano performances and generate coherent continuations. Our baseline model follows a simple two-stage architecture inspired by the AudioLM pipeline. First, we use the pretrained SoundStream encoder and decoder (Shively 2022) to convert raw audio waveforms into discrete acoustic tokens, capturing temporal dependencies and providing higher reconstruction quality than other codecs, such as Opus and EVS. SoundStream was used for its discrete sequence representation, which matches our transformer prediction format.

The Soundstream encoder consists of a series of convolutional and residual blocks with strided convolutions which downsample the input. Then a Residual Vector Quantizer (RVQ) discretizes the continuous embeddings into quantized codes which serves as input to a Transformer-based language

model. The Transformer processes the quantized codes and is trained to predict the next token in the sequence, enabling autoregressive continuation generation. During training we optimize negative log-likelihood (NLL) loss in alignment with related works. In the baseline, the model takes in only the first quantized level out of eight, which only captures the high-level structure of the piano piece.

The transformer operates in the discrete token space, meaning it never processes the raw audio directly. At inference time, the predicted tokens are decoded back into audio using the SoundStream decoder.

To handle the high temporal resolution of raw audio, we pre-process the 2018 subset of the MAESTRO dataset by windowed chunking. Each audio file is segmented into overlapping 15-second chunks and encoded into discrete tokens. Overlaps between adjacent chunks preserve temporal continuity to reinforce smooth generations. Token sequences from all chunks are concatenated and divided into fixed-length sliding windows of 2048 tokens with 50% overlap, producing continuous tokens that maintain the sequential structure required for next-token prediction. A full depiction of the baseline architecture is shown in Figure 1

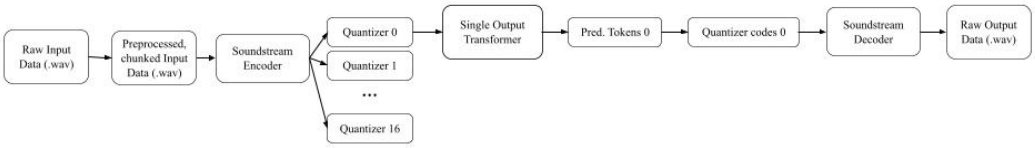

Figure 1: Baseline Architecture.

3.2 Novel Architecture

Building off of the baseline model, our novel model replaces the original transformer with a new transformer architecture that produces multiple outputs. We also introduce a new loss metric for training our model.

Form the baseline model, we maintain the Soundstream encoder and decoder to convert between raw audio and discrete, quantized tokens. As mentioned, there exist a total of 16 levels of quantized tokens, of which we used the primary level for the baseline architecture. Soundstream uses Residual Vector Quantization in which the residuals from one quantization level are used to develop the next quantization level, illustrated in Figure 2.

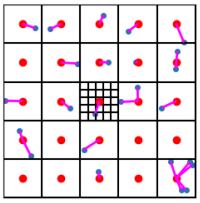

Figure 2: Illustration of redisuals, shown as purple line segments connecting vectors (blue points) with their nearest centroids (red points) (Hawley 2023).

In the novel approach, we train a transformer capable of producing outputs across multiple quantization levels, allowing it to reconstruct the vectors with finer-grained detail. In this setup, the model conditions on the first four levels of quantized tokens and predicts the next four levels at each time step, as illustrated in Figure 4.

Each input sequence is represented as a tensor $X \in \mathbb{Z}^{B \times T \times Q}$, where $B = 8$ is the batch size, T is the sequence length, and $Q = 4$ is the number of quantization levels. Each token value is an integer representing its code in a codebook of size 1024.

The transformer embeds each token via an embedding matrix of shape $E_{emb} \in \mathbb{R}^{V \times D}$, where $D = 512$ is the embedding dimension. These embeddings have dimension $B \times T \times Q \times D$, but are flattened along the quantizer dimension, such that information about all levels is available for

training at each time step. To incorporate sequential information, we add a positional encoding via the position matrix $E_{pos} \in \mathbb{R}^{T \times D}$.

The transformer consists of 6 layers, each with multi-head self-attention with 8 heads. It implements residual connections, layer normalization, dropout (with $p = 0.1$), and a final feedforward network with dimension $D_f = 2048$. We add a causal mask which simply prevents attention on future timesteps.

The output of the transformer is normalized and passed through a linear layer to produce logits corresponding to code from the codebook for each quantizer. The final tensor $O \in \mathbb{V}^{B \times T \times Q \times V}$, where $\mathbb{V}$ is the set of all codebook values. Each slice along the quantizer dimension represents the predicted tokens for that quantization level at every time step. During training, tokens are compared with target tokens from the same level, enabling the model to learn level-specific features while still capturing cross-level dependencies through shared context. The predicted tokens are finally passed to the SoundStream decoder to generate the audio continuation.

Because of the complexity of learning long-range temporal structure in audio embeddings, employed PyTorch's One-Cycle Learning Rate scheduler. According to the PyTorch documentation, "The 1cycle policy anneals the learning rate from an initial learning rate to some maximum learning rate and then... to some minimum learning rate much lower than the initial learning rate" (PyTorch contributors 2025). This warm-up followed by smooth decay is known to improve convergence and reduce training instability. To take full advantage of this schedule, we extended training by an additional five epochs relative to the baseline.

The second major modification explored in our novel approach was the introduction of a Mel-spectrogram-based perceptual loss. Traditionally, Mel-spectrogram loss is applied to raw audio waveforms rather than discrete latent tokens. However, our architecture enables a pseudo-Mel loss by applying a differentiable Mel-spectrogram transform to a linear reconstruction of latent token embeddings. Specifically, both the predicted token sequence and the ground-truth token sequence are projected into a shared embedding space, converted into Mel-spectrograms, and compared using an L1 loss. The difference is depicted in Figure 3. This allows the model to optimize for

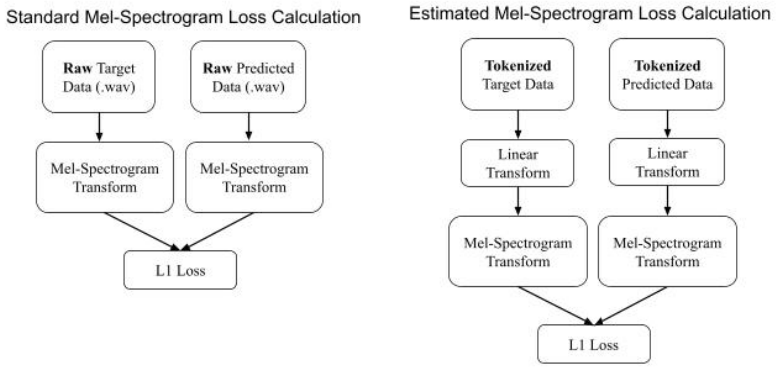

Figure 3: Mel Architecture.

perceptually relevant frequency-domain structure even without decoding back to waveform, reducing computational cost while encouraging musical coherence. This loss is then combined with the same NLL Loss that was used in the baseline model.

A full depiction of the novel architecture, as described, is shown in Figure 4. We lastly test a few lowpass filters to compensate for the lack of higher quantization levels. We based on qualitative tests, we set the lowpass filter to have a threshold of under 0.05 of the Nyquist frequency of the signal.

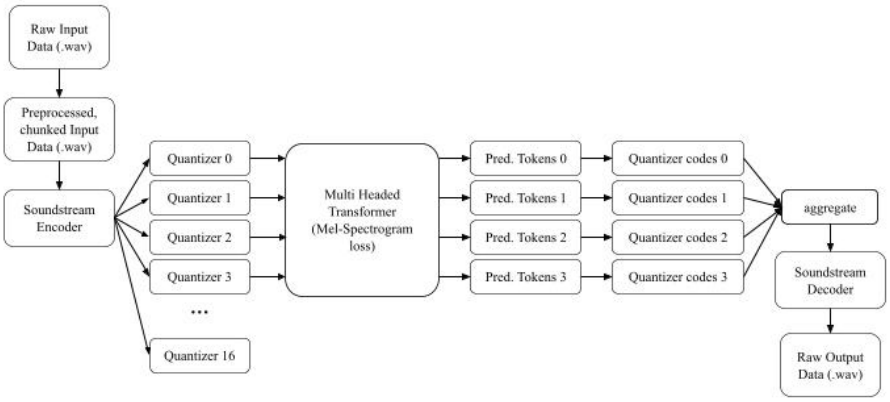

Figure 4: Novel Architecture.

4 Results

4.1 Baseline Results

As shown in Figure 5, the best training loss attained for the baseline was **2.6820** and the best validation loss attained for the baseline was **2.6265**.

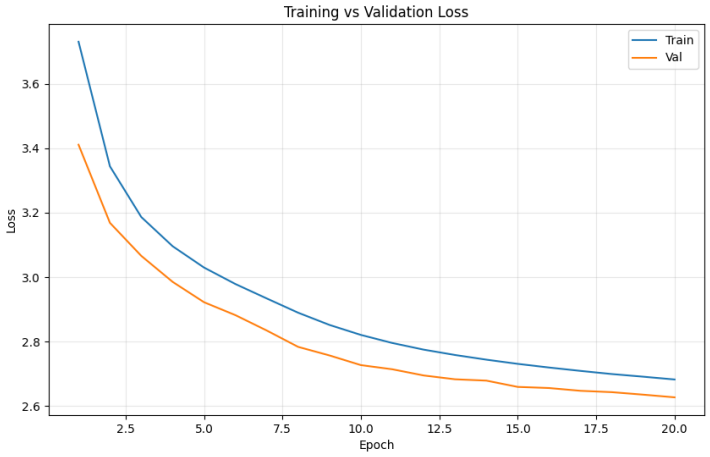

Figure 5: Baseline Training Loss.

Our baseline transformer model was trained for 20 epochs on the MAESTRO-2018 dataset tokenized using only the first SoundStream quantizer. As shown in the loss plots, both training and validation loss decrease smoothly and do not diverge, indicating stable optimization and no evidence of overfitting. The best validation loss (2.6265) was slightly lower than the best training loss (2.6820), which suggests that the model is regularized well, but also that it may still be underfitting and has not fully learned the distribution of musical structure in the dataset.

Although the loss curves indicate learning progress, the qualitative results do not align with the quantitative metrics. The generated audio is muffled, noisy, and lacks recognizable piano timbre. This outcome reveals an important limitation of the baseline design: loss on the first quantizer level token prediction does not correlate with perceptual audio quality. Since only the highest-level quantizer is modeled, the transformer learns coarse temporal structure but cannot recover harmonic content, dynamics, or spectral detail.

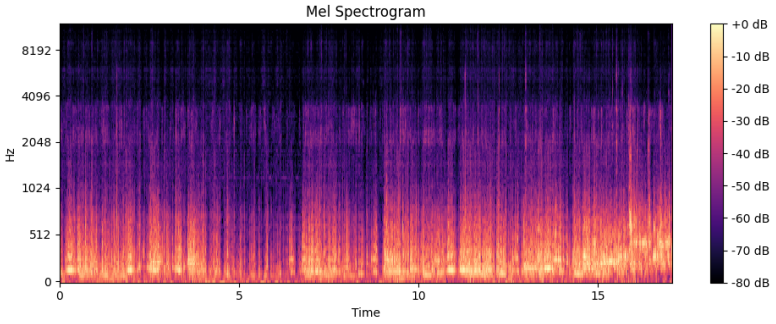

Figure 6: Baseline Mel Spectrogram of a Random Sample Window

The spectrogram of the generated audio confirms this gap between learned structure and audible fidelity. Unlike real piano audio, which shows clear harmonic bands and note onsets, the baseline output contains diffuse, low-frequency energy with no stable harmonics. This is consistent with prior findings in neural audio modeling: a single quantizer level is insufficient to reconstruct realistic sound, and multi-level token prediction is required for timbre fidelity.

4.2 Novel Results

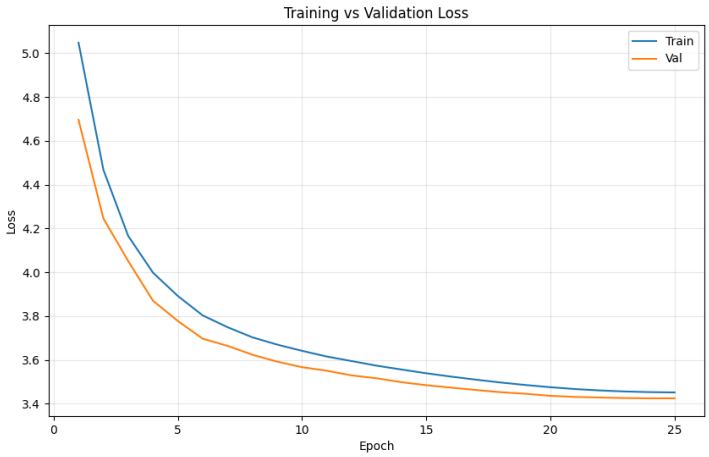

Figure 7: Novel Training Loss.

As shown in Figure 7, the best training loss attained for the novel model was **3.4515** and the best validation loss attained for the novel model was **3.4246**.

Our multi-output transformer was trained for 25 epochs on the same dataset and with the same tokenization pipeline as the baseline model. For a fair comparison, we report results in terms of cross-entropy loss (CE), the metric used for baseline evaluation. The loss curves show that training decreased smoothly and exhibited no divergence, indicating stable optimization despite the greater complexity of predicting four quantizer streams simultaneously. The fact that the best validation loss (3.4246) is slightly lower than the best training loss (3.4515) suggests mild underfitting and strong regularization, consistent with the One-Cycle LR schedule and the larger prediction head.

Since we introduced a new loss function for the novel model, we also evaluate the generated audio according to this metric. Figure 8 presents the Mel-Spectrogram corresponding to a selected generated window. Relative to the baseline spectrogram, the novel model exhibits clearer harmonic structure and smoother temporal continuity, as evidenced by a more even distribution across frequencies and stronger correlation across time. These differences indicate that the multi-quantizer prediction

objective, together with the Mel-based perceptual loss, produces representations that more closely resemble real piano recordings. Although the token-level cross-entropy loss remains higher than that of the baseline, the spectrogram provides qualitative evidence that the novel loss function improves perceptual audio characteristics. Compared to the baseline model, the generated audio is less muffled and the piano timbre is recognizable through the noise.

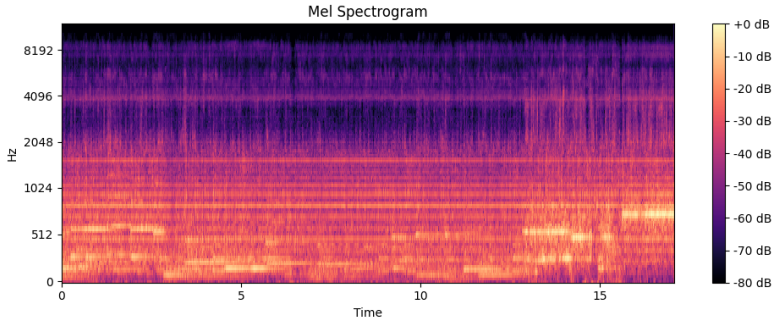

Figure 8: Novel Mel Spectrogram of a Random Sample Window.

5 Discussion and Analysis

Overall, the baseline training was technically successful (stable loss, no mode collapse, no overfitting), but perceptually unsuccessful. This result validates our hypothesis that token-level accuracy alone is not a reliable indicator of audio quality, and motivates the need for multi-quantizer level prediction and alternative loss function.

Results for the novel model show a different trend. Even though the training and validation losses were higher than those of the baseline model, the generated spectrograms indicate improved sound quality. The model produced clearer horizontal harmonic bands, smoother transitions between frames, and reduced high-frequency sounds. These improvements suggest that the multi-quantizer formulations allows the model to capture an improved representation of the SoundStream latent space, and that the Mel-Spectrogram objective provides a helpful guidance even when operating on token embeddings instead of raw audio. The Mel-spectrogram produced by our model, shown in Figure 9, exhibits similar overall structure to that of the Jukebox model in Figure 10. However, the Jukebox spectrogram shows noticeably smoother temporal transitions. The more abrupt transitions in our model's output are likely due to the absence of higher-level quantization levels. Since Jukebox implements a hierarchal structure, it is better fit to encode and retain finer-grained acoustic detail, including the subtle information that defines smooth transitions between frames.

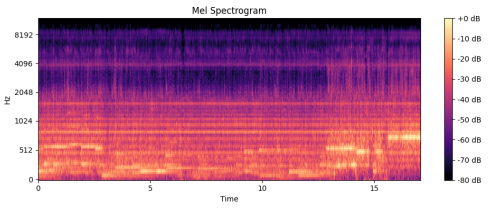

Figure 9: Novel Mel Spectrogram.

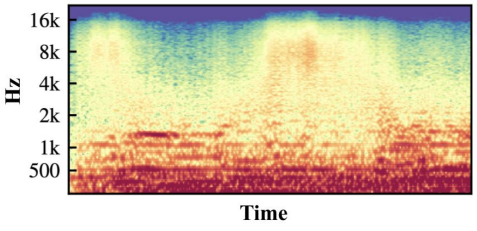

Figure 10: Jukebox Mel Spectrogram (Dhariwal et al. 2020). Stride length 128, no spectral loss.

In theory, splitting prediction by quantization level allows for fine-grained evaluation of different, important components of a musical piece. This is due to the nature of the Residual Vector Quantization used in Soundstream. As shown in Figure 11, more levels of quantization leads to lower reconstruction error.

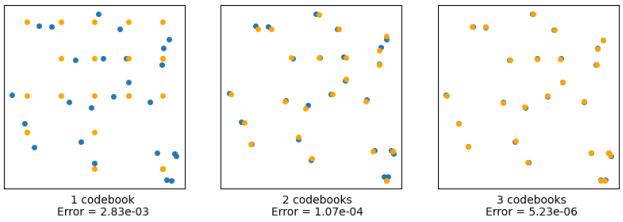

Figure 11: Reconstructing data (blue points) using multiple levels of RVQ codebooks (orange dots)(Hawley 2023).

However, in practice, using all 16 quantization levels was beyond the time and memory limitations of the resources available. Thus, the results suggest that all 16 levels quantization are important for generating coherent, clear audio and that the latent scape for audio vastly differs from the latent space for data such as text. The full audio file contained both the decoded seed audio and the decoded continuation. From human observation, even the decoded seed tokens for generation had a significant level of noise, implying that the remaining 12 quantization levels contribute significantly to the sound quality. Thus, one clear way to improve the model's performance would be to train a transformer that is able to predict all 16 levels of quantization in order to capture all details of the encoded audio.

When generating audio with a length longer than the duration of the seed audio, we observe that the audio slowly converges to noise. As tokens are conditioned on previously generated tokens, coherence decreases as more of the context consists of model-produced tokens. With an attention window of 2048 tokens, the effects become significant around 1000 tokens later. To mitigate this, we may consider implementing a different type of encoder which preserves long term features beyond the first quantization level of the Soundstream encoder. We also make the assumption that the open source implementation followed the convention of lower quantization levels encoding longer term structure while higher quantization levels encode more detail. Future work should incorporate all quantization levels or integrates a different encoder that better preserves long-term dependencies.

Comparing to results from existing models in Table 1, we see that our NLL loss trends larger than other models. We can attribute some of the performance to a lack of training resources. However, it is also important to note that though NLL is the most consistent metric used across literature and thus used for comparison, it may not be the most accurate indicator of audio quality. Most papers generally seek other metrics such as sWUGGY and sBLIMP (Borsos et al. 2022) or spectral convergence (Dhariwal et al. 2020) which are not consistent across the field.

Model	**NLL Loss**
WaveNet	1.151
VQ_VAE	1.172
Baseline	2.6265
Novel	3.4246

Table 1: Comparison of validation NLL Loss (Dieleman, Aäron van den Oord, and Simonyan 2018)

Further, the qualitative results suggest that training with Mel-spectrogram based loss aligns more closely with human perception than traditional reconstruction metrics. They also highlight the importance of multiple quantization levels in generating richer hierarchical codes which capture finer temporal details. These factors clearly contribute substantially to the perceived musical quality and overall listening experience.

The findings highlight that lower token-level loss does not necessarily correspond to better sounding audio, and optimizing directly for perceptual structure can improve output quality even when numerical losses appear worse. Our architectural and training modifications were designed primarily to enhance sound quality such as reducing noise, and strengthening harmonic structure. Once the audio quality is made clear and expressive, future work can shift toward modeling higher level attributes such as creativity, variation, and temperature during generation. These directions may allow the system to not only reproduce realistic audio, but also be able to generate novel piano continuations.

References

Borsos, Zalán et al. (2022). "AudioLM: A Language Modeling Approach to Audio Generation". In: *arXiv preprint arXiv:2209.03143*. URL: https://arxiv.org/abs/2209.03143.

Dhariwal, Prafulla et al. (2020). "Jukebox: A Generative Model for Music". In: *arXiv preprint arXiv:2005.00341*. arXiv:2005.00341 [eess.AS].

Dieleman, Sander, Aäron van den Oord, and Karen Simonyan (2018). "The challenge of realistic music generation: modelling raw audio at scale". In: *arXiv preprint arXiv:1806.10474*. arXiv:1806.10474 [cs.SD].

Hawley, Scott H. (2023). *Residual Vector Quantization*. https://drscotthawley.github.io/blog/posts/2023-06-12-RVQ.html. Published June 12, 2023. Accessed: 2025-12-04.

Hawthorne, Curtis et al. (2019). "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=r1lYRjC9F7.

Oord, Aaron van den et al. (2016). "WaveNet: A Generative Model for Raw Audio". In: *arXiv preprint arXiv:1609.03499*. arXiv:1609.03499 [cs.SD].

PyTorch contributors (2025). *torch.optim.lr_scheduler.OneCycleLR — PyTorch Documentation*. https://docs.pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.OneCycleLR.html. Accessed: 2025-12-04.

Shively, Hayden (2022). *SoundStream: Neural Audio Codec*. https://github.com/haydenshively/SoundStream. Commit fb850e5.