LadLe: Learning Audio Degradation to Lossless Enhancement Using CNN and Transformer Architectures

Jigyas Sharma
Math
University of Kansas
Lawrence, Kansas
engineer@ku.edu

***Abstract*—Audio compression has long been a trade-off between storage efficiency and sound fidelity, with lossy formats prioritizing compactness at the expense of subtle yet discernible details in audio quality. Motivated by the perceptual disparity between lossy and CD-quality audio, this research proposes a novel machine learning framework designed to approximate the data lost during compression. The system integrates a Convolutional Neural Network (CNN) encoder-decoder with a Transformer model to predict and restore missing audio features, leveraging the temporal structure of audio signals. Unlike traditional approaches, the model utilizes perceptual loss to align its reconstructions with human auditory preferences, ensuring a focus on the most salient aspects of audio fidelity. Preliminary evaluations demonstrate promising results in recreating high-fidelity audio that closely mirrors its original lossless form. This work not only strives to enhance the listening experience but also addresses the broader challenge of reconstructing lost data in resource-constrained scenarios, offering potential applications in fields such as audio restoration and digital archiving.**

I. Introduction

Since the introduction of Convolutional Neural Networks (CNNs), there have been significant advancements in machine learning, enabling models to address complex problems across a diverse range of domains. The versatility and effectiveness of CNNs are exemplified by their application to challenges such as image classification [1], natural language processing with visual inputs [2], and scientific data analysis, including breakthroughs in protein structure prediction [3].

Despite these substantial achievements, the adoption of CNNs for audio data processing remains comparatively under-explored, especially when contrasted with their extensive use in areas like image recognition and Monte Carlo approximations [4]. This work seeks to bridge this gap by investigating the applicability of CNNs to audio restoration tasks. Specifically, we present a method to reconstruct lossy audio segments into a lossless-like auditory experience using CNN-based architectures, enhanced by attention mechanisms and perceptual loss optimization.

This study introduces a 4-layer convolutional network to extract localized features from audio signals, complemented by a Transformer-based attention mechanism [5] to model global dependencies across the audio sequence. Furthermore, a perceptual loss function [6] is employed to ensure that the reconstructed audio aligns structurally with the original signal while maintaining its subjective auditory quality. The dataset is constructed by pairing lossless audio recordings with their lossy counterparts. Preprocessing involves dynamically segmenting the input into multiple time frames (e.g., 0.01-second, 0.1-second, and 1-second intervals) and applying various padding and resampling techniques to ensure alignment between corresponding lossless and lossy segments. This systematic approach ensures dataset consistency, providing a robust foundation for efficient model training.

The findings of this research demonstrate the potential of CNN-based architectures in audio restoration, offering broader implications for scenarios where degraded or incomplete data must be reconstructed to its original fidelity. This work also paves the way for expanded applications of CNNs in the audio domain and related fields.

II. Methodology

The methodology for this project encompasses three critical components: dataset creation, model components, and experimental settings. First, we construct a robust dataset by pairing lossless audio files with their corresponding lossy versions, ensuring alignment through preprocessing techniques like segmentation and resampling. The model architecture integrates a convolutional network and Transformer-based attention mechanism, designed to balance local feature extraction and global dependency modeling. Finally, the experimental settings define the hyperparameters, evaluation metrics, and training configurations necessary to achieve optimal performance. Each of these components is discussed in detail below.

A. Dataset Creation

To the best of my knowledge, no publicly available dataset exists that includes paired lossless and lossy audio samples suitable for this task. To address this gap, I constructed a custom dataset by extracting audio from compact discs (CDs), which store audio in a lossless format. The extracted audio was processed using a compression algorithm modeled after Spotify's method to generate corresponding lossy versions. The dataset comprises 115 songs with an average duration of 203 seconds, ensuring all audio files conform to a sampling rate of 44 kHz for the lossless versions and 32 kHz for the lossy versions, corresponding to 44,000 and 32,000 samples per second, respectively.

Figure 1 illustrates the difference between the waveform of a 1-second segment of a lossless audio sample and its corresponding lossy version. As shown, the lossy version exhibits a reduction in detail and amplitude precision, particularly in high-frequency regions. This highlights the challenges associated with reconstructing lossless-quality audio from lossy data, emphasizing the importance of the methods used for dataset alignment and preprocessing.

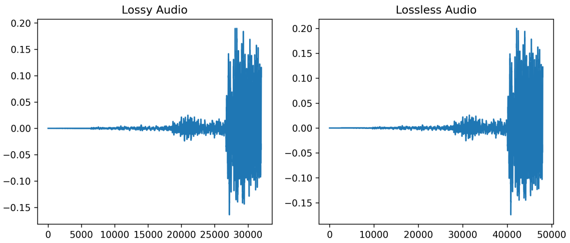


Fig. 1: Waveforms for a 1-second segment of lossless and lossy audio

Several approaches were explored for preprocessing the audio and creating the dataset:

1) **1-Second Segments with Zero Padding**: In the initial experiment, each song was segmented into 1-second intervals. Since the lossless audio contains 44,000 samples per second and the lossy audio contains 32,000 samples, the lossy segments were padded with 6,000 zeros at the beginning and end of the segment to match the dimensionality of the lossless data. This resulted in feature vectors with 44,000 neurons for each segment. A total of 25,323 segment pairs were generated in this setup.
2) **0.01-Second Segments with Distributed Zeros**: In the next approach, audio files were segmented into smaller intervals of 0.01 seconds. This resulted in lossless segments containing 440 samples and lossy segments containing 320 samples. To ensure alignment, zeros were distributed evenly between the samples of the lossy audio instead of padding only at the start and end. This method retained dimensional consistency while reducing the segment size for finer-grained analysis.
3) **0.1-Second Segments with Distributed Zero Padding**: Extending the above method, audio was segmented into 0.1-second intervals, resulting in 4,400 samples for the lossless audio and 3,200 samples for the lossy audio. Again, zeros were distributed evenly across the lossy segments to match the lossless dimensionality.
4) **0.1-Second Segments with Resampling**: In the final approach, the lossy audio was resampled to 44 kHz using the Torchaudio library, ensuring that both lossless and lossy audio segments shared the same sample rate. This eliminated the need for zero padding, creating a direct alignment between the two versions. The audio was then segmented into 0.1-second intervals for dataset creation.

The resulting dataset includes multiple variations to evaluate the impact of different preprocessing strategies on model performance. The results of these experiments are discussed in detail in Section IV.

B. Model Components

The proposed model consists of three key components: a CNN encoder, a Transformer-based feature processor and CNN decoder. Each component plays a crucial role in ensuring the effective restoration of lossy audio segments into a lossless-like quality. The functionality and design of these components are outlined below.

1) **CNN Encoder**: The CNN encoder is responsible for extracting low-level and intermediate-level features from the input audio signal. It consists of a series of 1-dimensional convolutional layers, each followed by a ReLU activation function.

The encoder progressively increases the feature dimensionality through its layers, starting from the raw input with two channels (stereo audio). By the final layer, the encoder produces a feature map of high dimensionality that represents the audio signal in a compact and informative manner. The hierarchical structure of the encoder allows the model to capture both local and global patterns in the audio data.

2) **Transformer**: The Transformer module processes the encoded feature map to model long-range dependencies and contextual relationships across the audio sequence. It employs multi-head self-attention mechanisms to capture complex relationships between time steps, enhancing the encoder's output with global context. The module consists of multiple Transformer encoder layers, each comprising self-attention and feed-forward submodules. The inclusion of the Transformer ensures that the model can effectively account for temporal dependencies, which are critical for audio reconstruction tasks.

3) **CNN Decoder**: The CNN decoder reconstructs the enhanced audio signal from the high-dimensional feature map produced by the Transformer. Using a series of transposed convolutional layers (also known as deconvolutions), the decoder progressively reduces the dimensionality of the feature map, transforming it back into the original input shape. Each layer is followed by a ReLU activation, except for the final layer, which uses a Tanh activation to constrain the output within a normalized range. The decoder's design complements the encoder, enabling the model to restore both fine-grained details and the overall structure of the audio signal.

C. Experimental Settings

The experimental setup consisted of carefully designed configurations to ensure consistent evaluation and reproducibility of results. The dataset was split into training and testing subsets in an 80:20 ratio, ensuring no overlap between the subsets.

The model was trained on a workstation equipped with an NVIDIA A100 with 40GB VRAM. PyTorch (version 2.5.1) was used for model implementation, with Torchaudio handling audio preprocessing. The Adam optimizer was used with a learning rate of 1×10^{-4} and a weight decay of 1×10^{-5}, while the batch size was set to 4 to balance memory usage and convergence. Training was conducted for a maximum of 20 epochs. Additionally, the learning rate was reduced by a factor of 0.5 if stagnation persisted for 5 epochs.

The model's performance was evaluated using two key metrics: the **Mean Squared Error (MSE)**, which quantifies the reconstruction loss between the predicted and target audio signals, and the **Perceptual Loss**, which measures high-level feature similarity using a pre-trained feature extractor. These metrics ensure that the reconstructed audio is assessed both in terms of numerical fidelity and perceptual quality. The experimental design allows for a robust comparison of results across different preprocessing strategies and model configurations. The outcomes are analyzed in Section IV.

III. Model Architecture

The proposed model is designed to restore lossy audio segments to a lossless-like quality. The architecture integrates convolutional feature extraction, Transformer-based attention mechanisms, and convolutional reconstruction, with a perceptual loss function to enhance both numerical and perceptual fidelity. The model consists of three primary components: a CNN encoder, a Transformer module, and a CNN decoder. Additionally, the perceptual loss incorporates a feature extractor, binding high-level feature similarity into the optimization process. A detailed description of each component and its mathematical formulation follows.

A. LadLe Architecture

The *LadLe* operates as a sequence-to-sequence model for audio signals. The model processes an input tensor $x \in \mathbb{R}^{B\times 2\times T}$, where B denotes the batch size, 2 represents the stereo audio channels, and T is the number of time steps. The architecture is composed of the following stages:

1) **CNN Encoder**: The encoder extracts hierarchical feature representations from the raw input signal using a sequence of four 1-dimensional convolutional layers. Each layer is parameterized by a kernel size of 3 and employs padding to preserve the temporal resolution. The operation of the encoder is defined as:

$$h = f_{\text{enc}}(x; \theta_{\text{enc}}),$$

where $h \in \mathbb{R}^{B\times d\times T}$ is the encoded feature map, d is the number of output channels of the final convolutional layer, and θ_{enc} represents the encoder's learnable parameters.

2) **Transformer Module**: The Transformer processes the encoded feature map h to model long-range dependencies and temporal relationships across the audio sequence. It consists of L Transformer encoder layers, each employing multi-head self-attention and feed-forward networks. The Transformer operates on permuted input $h' \in \mathbb{R}^{B \times T \times d}$, ensuring compatibility with its expected input format. The enhanced feature map is computed as:

$$h'' = f_{\text{trans}}(h'; \theta_{\text{trans}}),$$

where $h'' \in \mathbb{R}^{B \times T \times d}$ is the output of the Transformer and θ_{trans} denotes its learnable parameters.

3) **CNN Decoder**: The decoder reconstructs the time-domain audio signal from the enhanced feature map. It consists of four transposed convolutional layers that progressively reduce the feature dimensionality and restore the temporal resolution. The reconstructed audio $\hat{x} \in \mathbb{R}^{B \times 2 \times T}$ is obtained as:

$$\hat{x} = f_{\text{dec}}(h''; \theta_{\text{dec}}),$$

where θ_{dec} represents the decoder's learnable parameters. A Tanh activation function in the final layer ensures the output values are normalized.

B. Perceptual Loss Function

To ensure that the restored audio not only approximates the original signal numerically but also preserves its perceptual characteristics, a perceptual loss function is employed. The perceptual loss combines two terms:

1) **Reconstruction Loss**: The reconstruction loss measures the mean squared error (MSE) between the predicted and target audio:

$$\mathcal{L}_{\text{recon}} = \frac{1}{N} \sum_{i=1}^{N} (\hat{x}_i - x_i)^2.$$

2) **Perceptual Loss**: The perceptual loss compares high-level feature representations of the predicted and target audio. These features are extracted using a pre-trained feature extractor ϕ, parameterized to capture perceptually relevant characteristics. The loss is computed as:

$$\mathcal{L}_{\text{perc}} = \frac{1}{M} \sum_{j=1}^{M} \|\phi(\hat{x})_j - \phi(x)_j\|_2^2,$$

where M is the number of feature dimensions in the extracted representation.

The total loss function is a weighted combination of these two terms:

$$\mathcal{L} = \lambda_{\text{recon}} \mathcal{L}_{\text{recon}} + \lambda_{\text{perc}} \mathcal{L}_{\text{perc}},$$

where λ_{recon} and λ_{perc} control the contributions of the reconstruction and perceptual losses, respectively.

C. Role of the Feature Extractor

The feature extractor ϕ is a critical component of the perceptual loss, designed to extract representations that capture human-perceived audio quality. In this implementation, a lightweight CNN model is employed, consisting of two convolutional layers with ReLU activations. By focusing on high-level abstractions, the feature extractor ensures that the model learns to replicate perceptual features that are vital for audio quality, such as timbre and clarity, which may not be directly reflected in the numerical reconstruction loss.

IV. Experimental Results

This section evaluates the performance of the *LadLe* framework across the four preprocessing approaches detailed in Section II-A. Given the multidimensional nature of audio quality assessment, the evaluation incorporates both quantitative and qualitative analyses. Quantitatively, the performance is measured using the mean squared error (MSE) and perceptual loss metrics, which capture the reconstruction accuracy and perceptual fidelity of the restored audio, respectively. Qualitatively, subjective evaluations were gathered from two groups: four professional audio engineers and four amateur audiophiles. Participants rated the reconstructed audio based on timbre, clarity, and overall quality. Additionally, subjective ratings quantified the improvement over the original lossy audio and the perceptual difference between the restored and lossless audio. A scale of -5 to +5 was employed, where 0 indicates no improvement, -5 represents significantly worse than the lossy input, and +5 corresponds to indistinguishable from the lossless ground truth.

Table I provides a comprehensive summary of both quantitative metrics and qualitative feedback for the four preprocessing approaches.

A. Analysis of Results

The results reveal distinct patterns in the performance of the *LadLe* framework across different preprocessing approaches. Quantitative metrics such as MSE and perceptual loss provide a numerical assessment of reconstruction fidelity, while the qualitative ratings reflect

Preprocessing Approach	**MSE**	**Perceptual Loss**	**Improvement over Lossy (-5 to 5)**	**Quality Difference to Lossless (-5 to 5)**
1-Second with Zero Padding	0.003	0.031	2.7	1.8
0.01-Second with Distributed Zero Padding	0.001	0.098	-5.0	-5.0
0.1-Second with Distributed Zero Padding	0.005	0.063	1.2	-1.9
0.1-Second with Resampling	**0.008**	**0.029**	**3.8**	**2.9**

TABLE I: Quantitative and qualitative evaluation of preprocessing approaches.

subjective perceptions of improvement and closeness to the lossless audio. The observed trends are as follows:

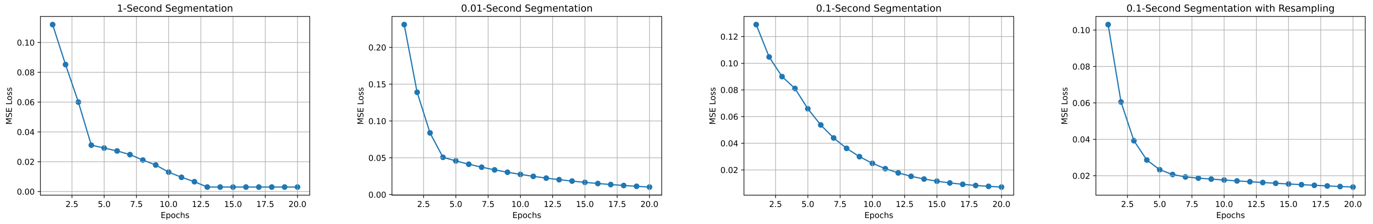
- **1-Second with Zero Padding**: This approach achieved a relatively low perceptual loss of 0.031, comparable to the best-performing approach, but its MSE of 0.003 suggests a moderate degree of reconstruction fidelity. In qualitative evaluations, it was rated positively for improvement over the lossy input (2.7) but showed limited perceptual similarity to the lossless reference (1.8).
- **0.01-Second with Distributed Zero Padding**: Despite achieving the lowest MSE (0.001), this approach exhibited the highest perceptual loss (0.098), indicating potential difficulties in capturing perceptual fidelity. Qualitative evaluations highlighted significant issues, with participants rating it at the lowest possible scores for both improvement over lossy audio (-5.0) and quality difference to lossless (-5.0), suggesting a negative auditory impact.
- **0.1-Second with Distributed Zero Padding**: This approach demonstrated moderate performance in both MSE (0.005) and perceptual loss (0.063). Qualitative evaluations revealed mixed results, with a slight improvement over the lossy input (1.2) but a perceptual quality rating of -1.9 relative to the lossless reference, indicating some perceptual degradation.
- **0.1-Second with Resampling**: While this approach did not achieve the lowest MSE (0.008), it recorded the lowest perceptual loss (0.029), indicating strong perceptual fidelity. Qualitative evaluations rated it as the best-performing approach, with scores of 3.8 for improvement over lossy audio and 2.9 for perceptual similarity to the lossless reference, reflecting favorable subjective impressions.

The combined quantitative and qualitative evaluations reveal significant insights into the *LadLe* framework's performance in restoring high-quality audio from lossy segments. A notable observation is the discrepancy between quantitative metrics and qualitative feedback. While approaches like 0.01-second segmentation achieved the lowest MSE and 1-second segmentation had competitive perceptual loss, qualitative analysis indicated that these approaches often resulted in degraded perceptual quality. This highlights that numerical convergence in metrics such as MSE or perceptual loss does not necessarily translate to better reconstructed audio. The resampling-based approach, despite not achieving the lowest MSE, consistently performed best in qualitative evaluations, demonstrating the importance of perceptual aspects in audio restoration. These results underline the critical role of combining both quantitative and qualitative assessments to gain a comprehensive understanding of model performance.

V. Conclusion

This paper explores the potential of CNN-based architectures in the domain of audio processing, with a focus on restoring audio quality from lossy segments. The proposed framework demonstrates its ability to enhance audio quality effectively, achieving reductions in both MSE and perceptual loss. These results highlight the capability of CNNs to address challenges in audio restoration while maintaining computational efficiency.

The findings presented in this work underscore a promising research direction for the application of universal approximation methods in audio restoration and processing. The experiments presented in this paper aim to serve as a baseline for evaluating future advancements in this domain, providing a foundation for exploring more sophisticated techniques and architectures. The study contributes to the expanding research on audio restoration, encouraging further exploration into the capabilities of deep learning models for audio processing tasks.



(a) 1-Second

(b) 0.01-Second

(c) 0.1-Second

(d) 0.1-Second with Resampling

Fig. 2: Convergence of MSE Loss for Different Preprocessing Approaches.

VI. Limitations and Future Work

The proposed framework exhibits several limitations that warrant further investigation. One significant challenge arises from the segmentation methods employed. When audio data was divided into 0.01-second pairs, the MSE loss decreased significantly; however, the reconstructed audio contained substantial noisy signals. Similar trends were observed for other segmentation approaches, including padding-based methods. Regardless of whether zeros were padded uniformly across the segment or concentrated at the boundaries, the resulting reconstructions often introduced noise. The best results were achieved with resampling, where lossy audio sampled at 32,000 samples per second was resampled to 44,000 samples per second using Torchaudio. This highlights the importance of data alignment: the closer the numerical values of lossy and lossless audio during training, the better the model's performance. When forced to fit padded zeros, the model exhibited instability, leading to noise in the reconstructed audio. This limitation of introducing noisy signals during reconstruction significantly degrades the listening experience.

Another limitation lies in the loss function. Although the reported loss values were low, this was often due to significant variability in the similarity between lossy and lossless segments. For instance, padding-based approaches resulted in segments at the start and end of songs being highly dissimilar, while middle segments were often better aligned. This inconsistency reduced the model's ability to consistently reconstruct high-quality audio across all segments.

A computational limitation also constrained the experiments. Even with access to a GPU with 40GB of VRAM, the maximum batch size was limited to four, leading to rapid convergence of the loss within the first epoch due to frequent updates from minibatches. This constrained the model's ability to generalize effectively and highlights the need for memory optimization techniques to enable training with larger batch sizes.

Future research could address these limitations and explore new directions for improvement. One potential avenue is to combine MSE and perceptual loss in a weighted manner during backpropagation, providing the model with a more nuanced feedback signal that accounts for both structural accuracy and perceptual fidelity. Another promising approach involves leveraging spectral graph representations of audio and computing loss based on discrepancies between the spectral graphs of lossy and lossless audio. Finally, expanding the framework to incorporate sequential models, such as LSTMs [7], could enable the system to utilize temporal dependencies, allowing it to better reconstruct audio by considering contextual information from prior segments.

References

[1] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *International Conference on Learning Representations*, 2015.

[2] Y. Kim, “Convolutional neural networks for sentence classification,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751, 2014.

[3] J. Jumper *et al.*, “Highly accurate protein structure prediction with alphafold,” *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.

[4] P. Grohs, A. Jentzen, and D. Salimova, “Deep neural network approximations for monte carlo algorithms,” *arXiv preprint arXiv:1908.10828*, 2019.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

[6] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *European Conference on Computer Vision*, pp. 694–711, 2016.

[7] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, pp. 1735–80, 12 1997.