Audio Super Resolution

Akarsha Sehwag
2015010

Manik Arora
2015053

Shiven Mian
2015094

YS Ramya
2015117

1. Introduction

1.1. Problem Statement and Motivation

Given an audio signal S1 at a low resolution, the task is to generate a higher resolution version signal S2 of S1, where the sampling rate of S2 ¿ S1, using different ML models. This is similar to Image Super Resolution, where the task is to super-resolve the image at high upscaling factors.

Audio Super Resolution has practical applications in telephony, compression, and text-to-speech generation; and neither of us has worked with signals before, giving us a motivation to pick up this project, also whatsapps poor quality audio was a big motivator for us.

2. Related Work

Audio Super resolution using CNNs: https://arxiv.org/pdf/1708.00853.pdf
Image super resolution using GANs: https://arxiv.org/abs/1609.04802

R1-¿ VCTK Dataset: http://homepages.inf.ed.ac.uk/jyamagis/page3/page58/page58.html
R2-¿ Piano Dataset: https://archive.org/search.php?query=piano%20beethoven

There have been two known previous works in this problem. Kuleshov et al (2017) use a deep learning based approach (a deep ResNet) for Audio Super resolution. Another previous work (Dong et al (2015)) use a Dictionary Learning based approach. We aim to simulate the results of Kuleshov et al, along with trying a GAN based model and some simpler deep models.

We are using two datasets - VCTK Corpus[R1] and a Piano Dataset[R2]. The VCTK Dataset is a set of wav files, with 109 native speakers speaking 400 sentences each. All the audio has been recorded in identical conditions. And the Piano dataset contains publicly available Beethovans Sonatas, consisting of around 10 hours of music.

3. Dataset and Evaluation

3.1. Pre processing and Data Division

The wav files were sampled at required sampling rate (16000 in our case), and the values were stored as a float array. This was further used in creating its low resolution counterpart where we randomly removed the data after regular intervals and replaced them by zero values. The zero values were then interpolated from the resultant array to give the final float array representation of low resolution data. Hence, we have pairs of low resolution data (input) and high resolution data (true value output) for all the audio files. We will split the dataset as 88%-6%-6%. Therefore, for VCTK dataset, around 9 speakers will be used for testing and for Piano Dataset, out of 32 publicly available Beethovans Sonatas, we'll use 1 for testing.

3.2. Evaluation Metrics

The performance of the model can be evaluated using the following two features of the input audio sample, actual audio sample and generated super resolved sample

1. **SNR**: Signal to Noise ratio
Given a signal y, and an approximation of the same signal x, the Signal to Noise ratio is given by:

$$\mathrm{SNR}(x,y) = 10\log\frac{||y||_2^2}{||x-y||_2^2}.$$

Figure 1. Figure 2

Higher the SNR values, the clearer the sound
2. **LSR**: Log Spectral Distance
It's a distance measure to evaluate reconstruction quality of signals. It is given by: Where X and Xâre log-spectral

$$\mathrm{LSD}(x,y) = \frac{1}{L}\sum_{\ell=1}^{L}\sqrt{\frac{1}{K}\sum_{k=1}^{K}\left(X(\ell,k)-\hat{X}(\ell,k)\right)^2},$$

Figure 2. Figure 2

power magnitudes of y and x respectively. Lower the log spectral distance indicates matching frequency.

4. Analysis and Progress

4.1. Progress so far

We have created a three layered Auto Encoder which gives us decent results.

4.2. Challenges Faced

1. **Technical Challenges:** Library issues in HPC, therefore we have currently used partial set of the entire data. (2432 audio samples of 3 seconds)

2. Scaling and Shifts in Results

3. Accuracy and performance of the trained models.

4.3. Design Choices

1. **Model Selection:** We decided to try out both reconstruction (autoencoder) as well as generative approaches (LSTM, SRGAN, Hidden Markov Model and Resnet (Kuleshov et al)).

2. Final models will be selected by the most accurate solution obtained, as defined by our loss function. (Mean Squared Error Loss)

3. In our current approach, we have tested an Autoencoder-based reconstruction approach with the following hyperparameters:

3.1 Batch Size = 16 (vs 4 vs 1)

3.2 Epochs = 4 (vs 1)

3.3 Input — Layer 1 — (Activation: ReLU) — Layer 2 – (ReLU) — Layer 3 (ReLU) — Output Layer

4. Explanation for Hyperparameters Choice:

4.1 **Speed**- Although Speed is not the most important factor when working with high capability processors, for initial testing on local machine, speed is a limiting factor. For speed purposes, ReLU worked faster than Sigmoid, as is apparent, since Sigmoid uses exponential function.

4.2 **Accuracy** - We obtained more accurate results by using ReLU Activations in both layers as compared to other combinations of Sigmoid and ReLU.

4.3 **Convergence** - We used smaller batch sizes which helped us reach stable kernel values faster as compared to larger batch sizes.

4.4. Supporting Evidence

According to Fig1 and Fig2 which are float arrays of the low resolution and high resolution graphs plotted respectively, it is apparent that values can be predicted between data points of the low resolution signal to add additional data points to transform it into a higher resolution signal. Hence, generative models are appropriate to predict new values between data points. Autoencoders can be used for the same purpose by trying to reproduce the high resolution signal via the neural net from the input of lower resolution signal.

Our current trained model could be improved to achieve better accuracy. We need to explore more hyperparameters and analyse their effect on training. Our model seems not to be overfitting or underfitting the data.

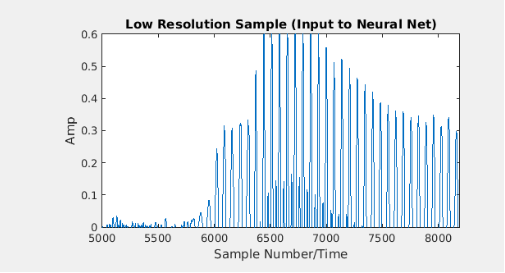

Figure 3. Figure 1

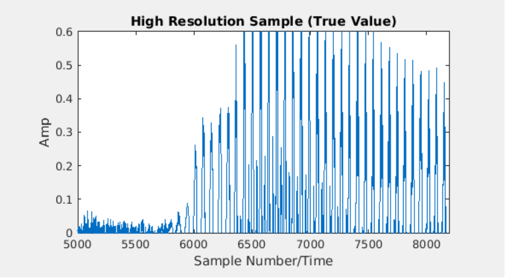

Figure 4. Figure 2

4.5. Data Domain Analysis

Audio data has many characteristics which are reflected in the plots. The data points become more dense near every crest and trough and the sample appears to have continuous behaviour since audio is a continuous signal, and not a discrete one. Since we have discretized this continuous signal, we obtain values with frequent alternating maxima and minima. Our predicted values, if correct, should remain within the immediate maxima and minima (or exceed very slightly), otherwise we can end up with irrelevant audio data points.

4.6. Results

We have obtained newly generated files of better resolution than the input. Fig 4 shows the newly obtained waveform of a testing data for instance. We have also calculated the 2 evaluation metrics for our current model as shown in Fig 5.

As can be seen, our reconstructed signal from the Autoencoder isnt a big improvement over the low resolution

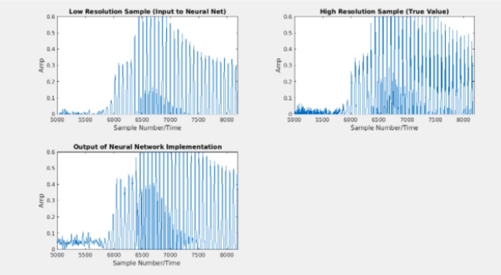

Figure 5. Figure 1

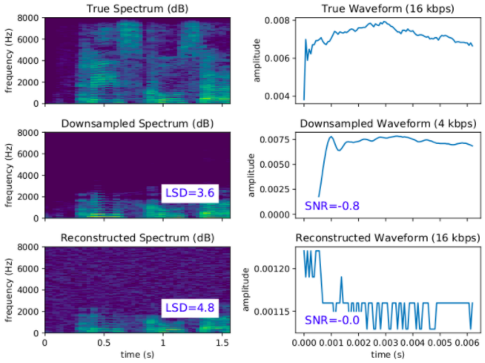

Figure 6. Figure 1

signal, which was what we expected. We expect the generative models to give significantly better results.

4.7. Interpretation of Results

The current results are satisfactory according to observation of the the generated waveforms which are similar to the high resolution audio waveforms visually. The quantitative measures show an increase in SNR as output of our model, which implies we have clearer audio, and increase in LSD which shows that our predicted values do not agree with the float values of our high resolution audio accurately, and improvement in model is required.

4.8. Insights from Analysis

Our current tests show that our next immediate aim with the current model is to predict values closer to the true values of the high resolution audio. We can do this by exploring more architectures and increasing the input values of our grid search. We have got clear audio in the current model which can be, in fact, used as an input to the neural network which successfully increases LSD. Hence, via a hybridization of the current and the model we aim to obtain, we can achieve a decrease in LSD and increase in SNR value.

4.9. Modification in project Proposal

We have selected the VCTK Corpus from the set of datasets provided in the proposal, and have included a new Piano Dataset as mentioned above. Also, instead of using a WGAN and VAE, we decided to try simpler models (LSTM and HMM) instead. The evaluation metrics remain the same as the proposal.

4.10. Future Analysis Pathway

We wish to check performance of simpler models on the Audio Super Resolution task, and compare them with state-of-the-art models. We then see how well a Generative Adversarial Network performs on this problem.

4.11. Member Roles

Shiven Mian: Data Preprocessing, Explore LSTM and SRGAN for Audio Resolution

Manik Arora: Data Preprocessing, Explore Auto-Encoder and Hidden Markov Model,

YS Ramya: Data Preprocessing, Implement Kuleshov et al. model (2017),

Akarsha Sehwag: Data Preprocessing, Explore Hidden Markov Models and CNN (Kuleshov et al).