Time Series Super Resolution with Temporal Adaptive Batch Normalization

Anonymous Author(s)
Affiliation
Address
email

Abstract

In many machine learning problems, data is naturally expressed as a time series. Here, we introduce a deep neural network architecture for reconstructing a high-resolution time series signal from low-resolution measurements, a task that we call time series super resolution. Central to our architecture is a novel temporal adaptive normalization layer that combines the strength of convolutional and recurrent approaches. We apply our model to diverse super resolution problems: audio super-resolution and the enhancement of functional genomics assays. In each case, our method significantly outperforms strong baselines, demonstrating its ability to solve practical problems in a wide range of domains.

1 Introduction

Deep neural networks have recently achieved remarkable successes on difficult problems in signal processing such as compression [17], super-resolution [1], compressed sensing [6], and many others. In many of these tasks, signals are naturally represented using time series. In this paper, we define a signal processing problem that often arises in practical applications called *time series super resolution*, and we propose a new deep neural network model for solving this problem. At a high level, time series super-resolution consists in reconstructing a high-resolution signal from low-resolution measurements, both of which are sequences sampled over a period of time.

This paper focuses on two time series super resolution problems. The first task is audio super-resolution, which involves reconstructing high-quality audio from a low-quality input containing only a fraction (15-50%) of the original time-domain samples. The second task is the reconstruction of high-quality measurements from experimental assays in genomics using lower-quality measurements; this process can significantly drive down the cost of genomics experiments. In this context, measurements at different positions along the genome correspond to different points in time.

We propose to solve these problems using a deep neural network architecture that combines convolutional and recurrent layers in a novel way. Central to our architecture is a new layer called temporal adaptive normalization, in which convolutional filters are adaptively modified (either turned on or off) based on long-range information captured by a recurrent network. More specifically, we parametrize the rescaling parameters of a batch normalization layer as in earlier work on image stylization [2, 7] and visual question answering [14]; in our paper, the parametrization is handled by a recurrent neural network. This allows us to combine the speed of convolutional models with the ability of recurrent models to handle long-range dependencies within the input time series.

Empirically, our method outperforms strong baselines on each time series super-resolution task, demonstrating its usefulness in a wide range of domains. Interestingly, the model is domain-agnostic, yet outperforms more specialized approaches that make use of domain-specific features.

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

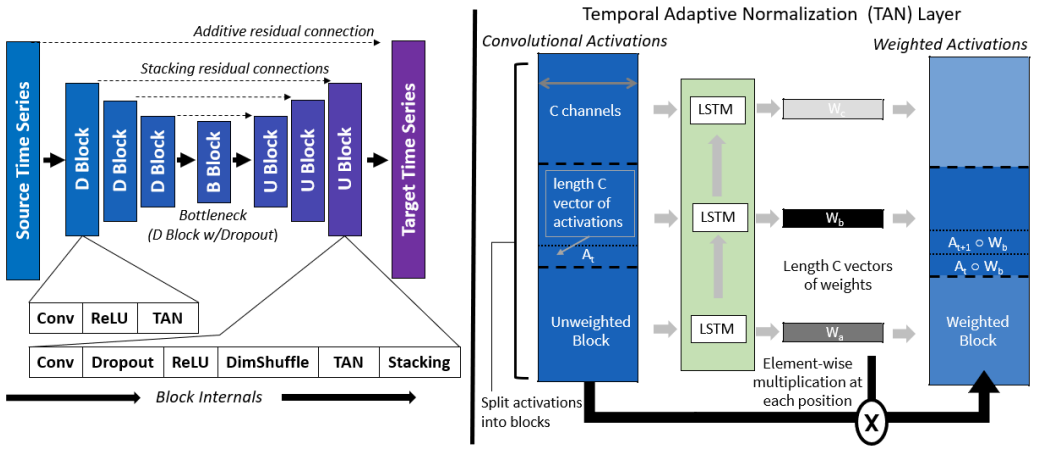

Figure 1: *Top left*: We propose a deep neural network architecture for time series super-resolution that consists of K downsampling convolutional blocks followed by a bottleneck layer and K upsampling blocks; features are reused via symmetric residual skip connections. *Bottom left*: Internal structure of each type of convolutional block. *Right*: Our new temporal adaptive normalization layer combines the strengths of convolutional and recurrent neural networks.

2 Time Series Super Resolution

We take a more general view of resolution than prior work on image super-resolution [1, 10]; its definition can vary across different application domains and can be very general. Specifically, we focus on two specific time series problems and use them as running examples throughout the paper.

Audio Super Resolution. Audio super resolution (also known as bandwidth extension; [3]) involves predicting a high-quality audio signal from a small fraction (15-50%) of its time-domain samples.Formally, given a low resolution signal $x = (x_{1/R_1}, ..., x_{R_1 T/R_1})$ sampled at a rate R_1/T (e.g. low-quality telephone call), our goal is to reconstruct a high-resolution version $y = (y_{1/R_2}, ..., y_{R_2 T/R_2})$ of x that has a sampling rate $R_2 > R_1$. We use $r = R_2/R_1$ to denote the *upsampling ratio* of the two signals. We thus expect that $y_{rt/R_2} \approx x_{t/R_1}$ for $t = 1, 2, ..., TR_1$.

Super Resolution of Genomics Experiments. Many genomics experiments can be seen as taking a real-valued measurement at every position of the genome; experimental results can therefore be represented by a time series. Measurements are generally obtained using a large set of probes (e.g., sequencing reads) that each randomly examine a different position in the genome; the genomic time series is an aggregate of the measurements taken by these probes. In this setting, super-resolution corresponds to reconstructing high-quality experimental measurements taken using a large set of probes from noisy measurements taken a small set of probes. This process can significantly reduce the cost of scientific experiments. In this paper, we focus on a particular genomic experiment called chromatin immunoprecipitation sequencing (ChIP-seq)[15].

3 A Deep Architecture Based on Temporal Adaptive Normalization

Super resolution naturally exhibits spatial invariance: the ideal reconstruction of a given subsequence does not depend of when it occurs in time. This naturally suggests a convolutional architecture for the problem. However, convolutions have a limited receptive field such that a subsequence of y depends on only a finite subsequence of x. On the other hand, recurrent neural networks (RNNs) have potentially infinite receptive fields, but can be slow and difficult to train on very long time series such as audio.

3.1 Temporal Adaptive Normalization

To address these limitations, we propose a new layer called Temporal Adaptive Normalization (TAN) that combines the strengths of convolutional and recurrent approaches. Our layer is based earlier work on adaptive batch normalization [14], in which the parameters γ, β of the affine normalization

Algorithm 1 Temporal Adaptive Normalization Layer.

Input: Tensor of activations $F \in \mathbb{R}^{N \times L \times C}$ from a 1D convolutional layer, where N, L, C are, respectively, the minibatch size, the spatial dimension, and the number of channels. Block length B. **Output:** Adaptively normalized tensor of activations $F' \in \mathbb{R}^{N \times L \times C}$.

1. Reshape F into a block tensor $F^{\text{blk}} \in \mathbb{R}^{N \times B \times L/B \times C}$, defined as $F^{\text{blk}}_{n,b,\ell,c} = F_{n,b \times \ell,c}$.

2. Compute sequence of normalizers $\gamma_b, \beta_b \in \mathbb{R}^C$ for $b = 1, 2, ..., B$ using an RNN: $(\gamma_b, \beta_b), h_b = \text{RNN}(F^{\text{blk}}_{\cdot,b,\cdot,\cdot}; h_{b-1})$ for $b = 1, 2, ..., B$ starting with $h_0 = \mathbf{0}$.

3. Compute normalized block tensor $F^{\text{norm}} \in \mathbb{R}^{N \times B \times L/B \times C}$ as $F^{\text{norm}}_{n,b,\ell,c} = \gamma_{b,c} \cdot F^{\text{block}}_{n,b,\ell,c} + \beta_{b,c}$.

4. Reshape F^{norm} into output $F' \in \mathbb{R}^{N \times L \times C}$ as $F'_{n,\ell,c} = F^{\text{norm}}_{n,\lfloor \ell/B \rfloor, \ell \bmod B, c}$.

in a batch norm layer are a function of an auxiliary input. In our case, the γ, β are conditioned on long range sequence information via an RNN.

Specifically, a TAN layer takes as input a tensor of activations $F \in \mathbb{R}^{N \times L \times C}$ from a 1D convolutional layer – where N, L, C are, respectively, the minibatch size, the 1D spatial dimension, and the number of channels – and applies a series of transformations. First, F is split along the time axis into blocks of length B to produce $F^{\text{blk}} \in \mathbb{R}^{N \times B \times L/B \times C}$. Intuitively, blocks correspond to regions along the spatial dimension in which the activations are closely correlated; for example, when processing audio, blocks could be chosen to correspond to audio samples that define the same phoneme. Next, we compute for each block b affine transformers γ_b, β_b using an RNN:

$$(\gamma_b, \beta_b), h_b = \text{RNN}(F^{\text{blk}}_{\cdot,b,\cdot,\cdot}; h_{b-1}) \text{ for } b = 1, 2, ..., B \text{ starting with } h_0 = \mathbf{0},$$

where h_b denotes the hidden state and $F^{\text{blk}}_{\cdot,b,\cdot,\cdot}$ is a 3D tensor obtained by fixing the second dimension to $b \in \{1, 2, ..., B\}$. In all our experiments, we use a convolutional LSTM.

Finally, activations in each block b are normalized by γ_b, β_b to produce a tensor F^{norm} defined as $F^{\text{norm}}_{n,b,\ell,c} = \gamma_{b,c} \cdot F^{\text{block}}_{n,b,\ell,c} + \beta_{b,c}$. Notice that each γ_b, β_b is a function of both the current and all the past blocks; hence they modulate activations using long-range signal from the RNN. In the audio example, the super resolution of a phoneme could depend on previous phonemes beyond the receptive field of the convolution; the RNN enables us to use this long-range information.

3.2 A Deep Neural Network Architecture for Time Series Super Resolution

The temporal adaptive normalization layer is a key part of our deep neural network architecture shown in Figure 1. The core of the model is formed by K successive downsampling and upsampling *layer blocks*.At a downsampling step, we halve the spatial dimension and double the filter size; during upsampling, this is reversed.We also add additional skip connections which stack the tensor of k-th downsampling features with the $(K-k+1)$-th tensor of upsampling features. In order to increase the time dimension during upscaling, we have implemented a one-dimensional version of the Subpixel layer of [16], which has been shown to be less prone to produce artifacts [13].

4 Experiments

4.1 Audio Super-Resolution

Setup. We use the VCTK dataset [18] — which contains 44 hours of data from 108 different speakers — and a Piano dataset (10 hours of Beethoven sonatas [12]). We generate low-resolution audio signal from the 16 KHz originals by applying an order 8 Chebyshev type I low-pass filter before subsampling the signal by the desired scaling ratio. The SINGLESPEAKER task trains the model on the first 223 recordings of VCTK Speaker 1 (about 30 mins) and tests on the last 8 recordings. In the MULTISPEAKER task, we train on the first 99 VCTK speakers and test on the 8 remaining ones.

We compare our method relative to two baselines: a cubic B-spline — which corresponds to the bicubic upsampling baseline used in image super-resolution — and a dense neural network (DNN) based on the technique of Li et. al., 2015 [11], We instantiate our model with $K = 4$ blocks and train it for 400 epochs on patches of length 8192 (in the high-resolution space) using the ADAM optimizer with a learning rate of 3×10^{-4}. To ensure source/target series are of the same length, the source input is pre-processed with cubic upscaling. We adjust the TAN block length B so that L/B (the number of blocks) is always 32.

		SingleSpeaker				MultiSpeaker				Piano			
Ratio	Obj.	Spline	DNN	Conv	Full	Spline	DNN	Conv	Full	Spline	DNN	Conv	Full
$r = 2$	SNR	20.0	19.5	19.4	19.3	18.1	19.9	19.6	19.5	24.7	24.3	25.3	25.7
	LSD	3.5	3.7	3.2	2.7	4.4	3.6	3.1	1.8	3.5	3.4	2.0	1.5
$r = 4$	SNR	15.6	15.9	16.3	17.2	13.0	12.7	13.1	14.9	18.6	18.6	19.1	19.3
	LSD	5.6	4.9	3.6	3.7	8.0	5.8	3.5	2.4	5.8	5.2	2.2	2.2

Table 1: Accuracy evaluation of audio-super resolution methods (in dB) on each of the three super-resolution tasks at upscaling ratios $r = 2, 4$ and 8.

Metrics Given a reference signal y and an approximation x, the signal to noise ratio (SNR) is defined as $\text{SNR}(x, y) = 10 \log \frac{||y||_2^2}{||x-y||_2^2}$. The log-spectral distance (LSD) [5] measures the reconstruction quality of individual frequencies as follows: $\text{LSD}(x, y) = \frac{1}{L} \sum_{\ell=1}^{L} \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left(X(\ell, k) - \hat{X}(\ell, k) \right)^2}$, where X and $\hat{X}$ are the log-spectral power magnitudes of y and x, respectively. These are defined as $X = \log |S|^2$, where S is the short-time Fourier transform (STFT) of the signal. We use ℓ and k index frames and frequencies, respectively; in our experiments, we used frames of length 2048.

Evaluation The results of our experiments are summarized in Table 4.1. Our basic convolution architecture shows an average improvement of about 0.5 dB over the spline baseline and about 0.3 dB over the DNN baseline, with the strongest improvements at medium upscaling ratios (i.e., 4). Including the TAN layers improves performance by an additional 0.9 dB on average.

4.2 ChIP-Seq Experiments

We use histone ChIP-seq data from lymphoblastoid cell lines derived from several individuals of diverse ancestry [8], and on the following common histone marks: H3K4me1, H3K4me3, H3K27ac, H3K27me3, and H3K36me3. This dataset contains high-quality ChIP-seq data with a high sequencing depth; to obtain low-quality versions, we artificially subsample 1M reads for each histone mark (out of the full dataset of 100+M reads per mark).

	Pearson correlation		
	Input (noisy)	CNN	Ours
H3K4me1	0.48	0.79	0.81
H3K4me3	0.66	0.83	0.90
H3K27ac	0.59	0.85	0.89
H3K27me3	0.21	0.65	0.64
H3K36me3	0.44	0.88	0.90

Table 2: Pearson correlation of the model output and the high-quality ChIP-seq signal derived from an experiment with high sequencing depth. The CNN baseline is from (Koh et. al., 2016) [9].

Formally, given an input noisy ChIP-seq signal $X \in \mathbb{R}^{k \times T}$, where k is the number of distinct histone marks, and T is the length of the genome, we aim to reconstruct a high-quality ChIP-seq signal $Y \in \mathbb{R}^T$. We use the k low-quality signals as input and train a separate model for each high quality target mark. We use $B = 2$ and training windows of length 1000; all other hyper-parameters are as in the audio-super resolution task.

To evaluate our results, we measure Pearson correlation between our model output and the true, high-quality ChIP-seq signal [4]. Across all of the histone marks, the model output from an input of 1M sequencing reads was equivalent in quality to signal derived from 10-20M reads, which is a significant efficiency gain.

5 Conclusion

In summary, our work defined time series super resolution, a task in which we reconstruct a high-resolution time series from low-quality samples. We introduce a neural architecture for this problem that is based on a new temporal adaptive normalization layer. We demonstrate our model's effectiveness in three diverse domains; our results have applications in text-to-speech generation, and can be used to reduce the cost of genomics experiments. We hope our model will be applied to new problems in science and engineering.

References

[1] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(2):295–307, February 2016.

[2] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *CoRR, abs/1610.07629*, 2(4):5, 2016.

[3] Per Ekstrand. Bandwidth extension of audio signals by spectral band replication. In *in Proceedings of the 1st IEEE Benelux Workshop on Model Based Processing and Coding of Audio (MPCA'02*. Citeseer, 2002.

[4] Jason Ernst and Manolis Kellis. Large-scale imputation of epigenomic datasets for systematic annotation of diverse human tissues. *Nature Biotechnology*, 33(4):364–376, 2 2015.

[5] Augustine Gray and John Markel. Distance measures for speech processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(5):380–391, 1976.

[6] Aditya Grover and Stefano Ermon. Amortized variational compressive sensing. 2018.

[7] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *CoRR, abs/1703.06868*, 2017.

[8] Maya Kasowski, Sofia Kyriazopoulou-Panagiotopoulou, Fabian Grubert, Judith B Zaugg, Anshul Kundaje, Yuling Liu, Alan P Boyle, Qiangfeng Cliff Zhang, Fouad Zakharia, Damek V Spacek, Jingjing Li, Dan Xie, Anthony Olarerin-George, Lars M Steinmetz, John B Hogenesch, Manolis Kellis, Serafim Batzoglou, and Michael Snyder. Extensive variation in chromatin states across humans. *Science (New York, N.Y.)*, 342(6159):750–2, 11 2013.

[9] Pang Wei Koh, Emma Pierson, and Anshul Kundaje. Denoising genome-wide histone chip-seq with convolutional neural networks. *bioRxiv*, page 052118, 2016.

[10] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. *CoRR*, abs/1609.04802, 2016.

[11] Kehuang Li, Zhen Huang, Yong Xu, and Chin-Hui Lee. Dnn-based speech bandwidth expansion and its application to adding high-frequency missing features for automatic speech recognition of narrowband speech. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[12] Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model, 2016. cite arxiv:1612.07837.

[13] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016.

[14] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. *CoRR*, abs/1709.07871, 2017.

[15] Consortium Roadmap Epigenomics. Integrative analysis of 111 reference human epigenomes. *Nature*, 518(7539):317–330, 2 2015.

[16] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. pages 1874–1883, 2016.

[17] George Toderici, Sean M O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. *arXiv preprint arXiv:1511.06085*, 2015.

[18] Junichi Yamagishi. English multi-speaker corpus for cstr voice cloning toolkit, 2012. *URL http://homepages. inf. ed. ac. uk/jyamagis/page3/page58/page58. html*.