Deep Learning for Audio

Lecture 6

Pavel Severilov

AI Masters

2023

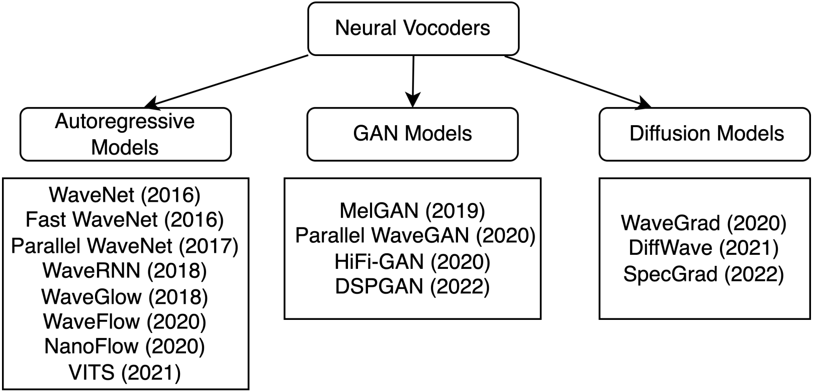
Outline

1. Neural Vocoders
2. WaveNet
3. Parallel WaveGAN
4. DiffWave

Outline

1. Neural Vocoders
2. WaveNet
3. Parallel WaveGAN
4. DiffWave

Neural Vocoders



Neural Vocoders

Model type	**Model**	**MOS on LJ Speech**
Autoregressive	WaveNet	3.68
	WaveRNN	3.96
GAN	MelGAN	3.73
	Parallel WaveGAN	3.99
Diffusion	WaveGrad	3.85
	DiffWave	**4.07**
	Griffin-Lim	3.68
	Ground Truth	**4.10**

AlBadawy et al., Vocbench: A Neural Vocoder Benchmark for Speech Synthesis, IEEE ICASSP, 2022

Outline

1. Neural Vocoders
2. WaveNet
3. Parallel WaveGAN
4. DiffWave

WaveNet

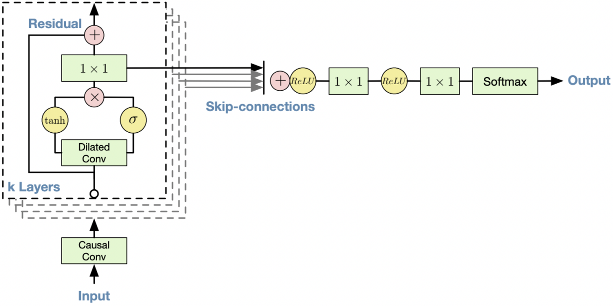


Figure: WaveNet architecture: uses **causal** dilated convolutions

- The joint probability of a waveform $x = \{x_1, \cdots, x_T\}$: $p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid x_{1:t-1})$
- Each conditional $p(x_t \mid x_{1:t-1})$ models the distribution for the timestamp t

Oord et al., WaveNet: A Generative Model for Raw Audio, 2016

Causal Convolution

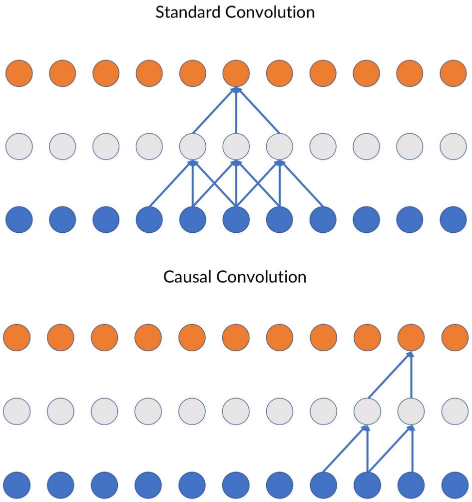


Figure: Standard vs causal convolutions. Causal makes convs autoregressive

Dilated Convolution

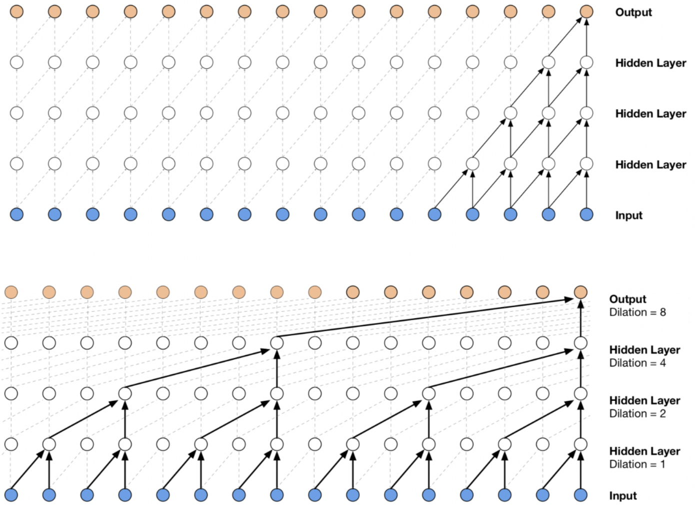
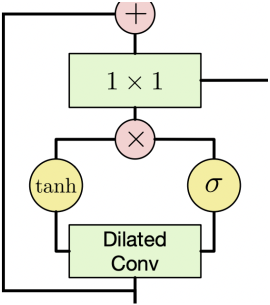


Figure: Non-dilated vs dilated causal convolutions. Dilated convs increase receptive fields

Conditional gated units

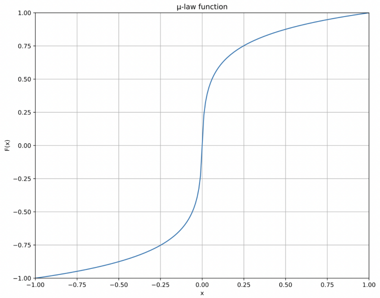


Gated activation unit as used in the gated PixelCNN + condition $\mathbf{y}$

$$\mathbf{z} = \tanh\left(W_{f,k} * \mathbf{x} + V_{f,k} * \mathbf{y}\right) \odot \sigma\left(W_{g,k} * \mathbf{x} + V_{g,k} * \mathbf{y}\right).$$

$*$ – convolution, $\odot$ – element-wise multiplication, $\sigma(\cdot)$ – sigmoid function, f and g – filter and gate, respectively, W – learnable convolution filter, V – learnable linear projection

Mu Law Encoding



- Raw audio $\sim$ 16-bit integer values $\Rightarrow$ softmax layer need to output 65,536 probabilities per timestep
- Solution: apply a μ-law transformation to the data, and then quantize it to 256 possible values:

$$f\left(x_t\right) = \operatorname{sign}(x_t)\frac{\ln\left(1+\mu\left|x_t\right|\right)}{\ln\left(1+\mu\right)}, \quad -1 < x_t < 1, \mu = 255$$

Outline

Parallel WaveGAN

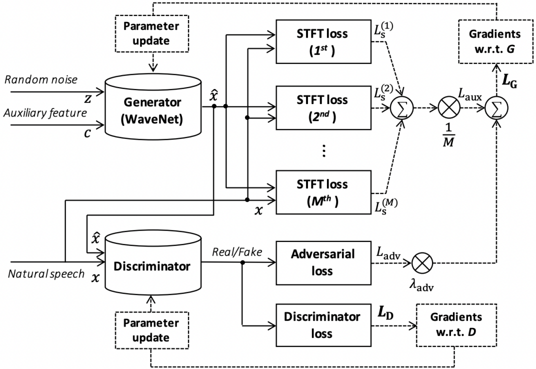
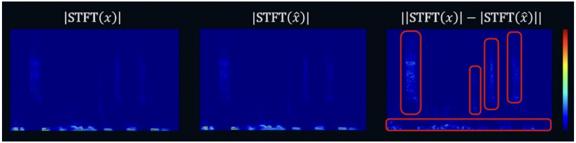


Figure: Parallel WaveGAN

Yamamoto et al., Parallel Wavegan: A Fast Waveform Generation Model Based on Generative Adversarial Networks with Multi-Resolution Spectrogram, IEEE ICASSP, 2020

STFT Loss



$$\mathcal{L}_{\mathrm{s}}(G) = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z}),\mathbf{x}\sim p_{data}}\left[\mathcal{L}_{\mathrm{sc}}(\mathbf{x},\hat{\mathbf{x}}) + \mathcal{L}_{\mathrm{mag}}(\mathbf{x},\hat{\mathbf{x}})\right]$$

$$\mathcal{L}_{\mathrm{sc}}(\mathbf{x},\hat{\mathbf{x}}) = \frac{\left\| |\mathrm{STFT}(\mathbf{x})| - \left|\mathrm{STFT}^{\mathrm{T}}(\hat{\mathbf{x}})\right| \right\|}{\| |STFT(\mathbf{x})| \|_F}$$

$$\mathcal{L}_{\mathrm{mag}}(\mathbf{x},\hat{\mathbf{x}}) = \frac{1}{N}\left\| \log|\mathrm{STFT}(\mathbf{x})| - \log|\mathrm{STFT}(\hat{\mathbf{x}})| \right\|_1$$

$$\mathcal{L}_{\mathrm{aux}}(G) = \frac{1}{M}\sum_{m=1}^{M}\mathcal{L}_{\mathrm{s}}^{(m)}(G), \mathrm{M} - \text{number of STFT losses}$$

Takaki et al., STFT Spectral Loss for Training a Neural Speech Waveform Model, IEEE ICASSP, 2019

GAN Loss

- Discriminator Loss:

$$\mathcal{L}_{\mathrm{D}}(G, D) = \mathbb{E}_{\mathbf{x} \sim p_{\mathrm{data}}}[(1 - D(\mathbf{x}))^2] + \mathbb{E}_{\mathbf{z} \sim N(0,I)}[D(G(\mathbf{z}))^2]$$

- Generator Loss

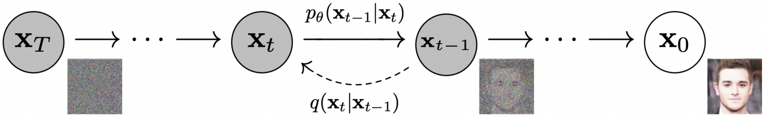
$$L_{\mathrm{adv}}(G, D) = \mathbb{E}_{\mathbf{z} \sim N(0,I)} \left[(1 - D(G(\mathbf{z})))^2\right]$$

$$L_{\mathrm{G}}(G, D) = L_{\mathrm{aux}}(G) + \lambda_{\mathrm{adv}} L_{\mathrm{adv}}(G, D)$$

Outline

1. Neural Vocoders
2. WaveNet
3. Parallel WaveGAN
4. DiffWave

Diffusion models idea



- **Diffusion probabilistic model**: parameterized Markov chain from data x_0 to the latent variable x_T

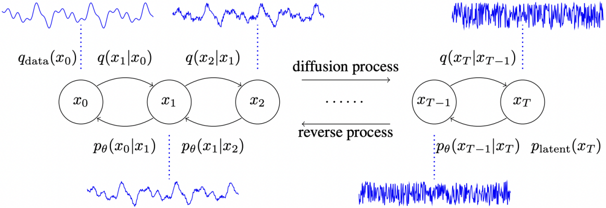
$$q(x_1, \cdots x_T|x_0) = \prod_{t=1}^{T} q(x_t|x_{t-1})$$

- **Reverse process**: Markov chain from x_T to x_0 parameterized by θ:

$$p_{\text{latent}}(x_T) = \mathcal{N}(0, I), \quad p_\theta(x_0, \cdots, x_{T-1}|x_T) = \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)$$

Ho et al., Denoising Diffusion Probabilistic Models, NeurIPS Proceedings, 2020

DiffWave



- **Sampling**: reverse process $x_T \sim \mathcal{N}(0, I)$, $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$ for $t = T, T-1, \cdots, 1$. x_0 – sampled data.
- **Training**: $p_\theta(x_0) = \int p_\theta(x_0, \cdots, x_{T-1}|x_T) \cdot p_{\text{latent}}(x_T) \mathbf{d}x_{1:T}$ (likelihood) is intractable to calculate $\Rightarrow$ model trained by maximizing its variational lower bound (ELBO):

$$\mathbb{E}_{q_{\text{data}}(x_0)} \log p_\theta(x_0) \geq$$

$$\geq \mathbb{E}_{q(x_0, \cdots, x_T)} \log \frac{p_\theta(x_0, \cdots, x_{T-1}|x_T) \cdot p_{\text{latent}}(x_T)}{q(x_1, \cdots, x_T|x_0)} := \text{ELBO}$$

Kong et al., DiffWave: A Versatile Diffusion Model for Audio Synthesis, ICLR, 2021

DiffWave: advantages

- **Non-autoregressive**: much faster than WaveNet
- **Compact model**: smaller footprint than flow-based models
- No auxiliary losses in training (e.g., spectrogram-based losses): **no mode collapse** like in GANs/VAEs