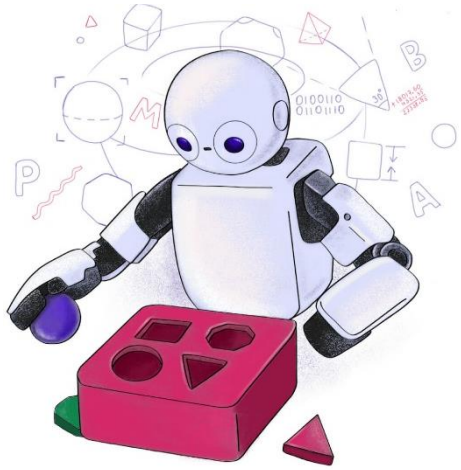
# TP558 - Tópicos avançados em aprendizado de máquina: *WaveNet*

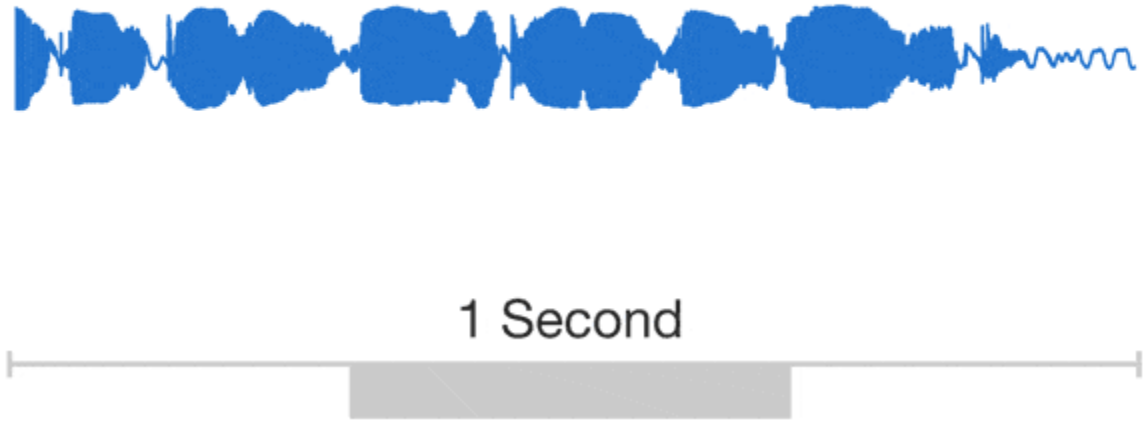


Inatel

Adrian A. C. Alanes
adrian@mtel.inatel.br

# Introdução

- O WaveNet, desenvolvido pela DeepMind em 2016, é um modelo generativo autoregressivo que produz o sinal de áudio diretamente na forma de waveform, amostra por amostra.
- Ao contrário dos sistemas TTS concatenativos e paramétricos, gera fala com qualidade muito mais natural e expressiva, aproximando-se da voz humana.
- O modelo consegue imitar múltiplos falantes, capturando sotaques, entonações e características específicas da voz.
- Suas aplicações vão além da fala: podem incluir síntese musical, efeitos sonoros e até tarefas de reconhecimento de fala.



# Fundamentação teórica

- **Modelagem autorregressiva**, cada amostra de áudio $x_t$ é prevista considerando todas as amostras anteriores.
- Isso significa que o sinal completo pode ser descrito como uma probabilidade conjunta, fatorada em uma sequência de probabilidades condicionais:

$$p(\mathbf{x}) = \prod_{t=1}^{T} p\left(x_t \mid x_1, \ldots, x_{t-1}\right)$$

# Fundamentação teórica

- **Convoluções causais**: garantem que o modelo respeite a ordem temporal: cada saída só pode depender das amostras atuais e passadas, nunca do futuro.
- Isso é essencial em tarefas de geração de áudio, pois o modelo precisa prever o próximo ponto da waveform de forma sequencial.

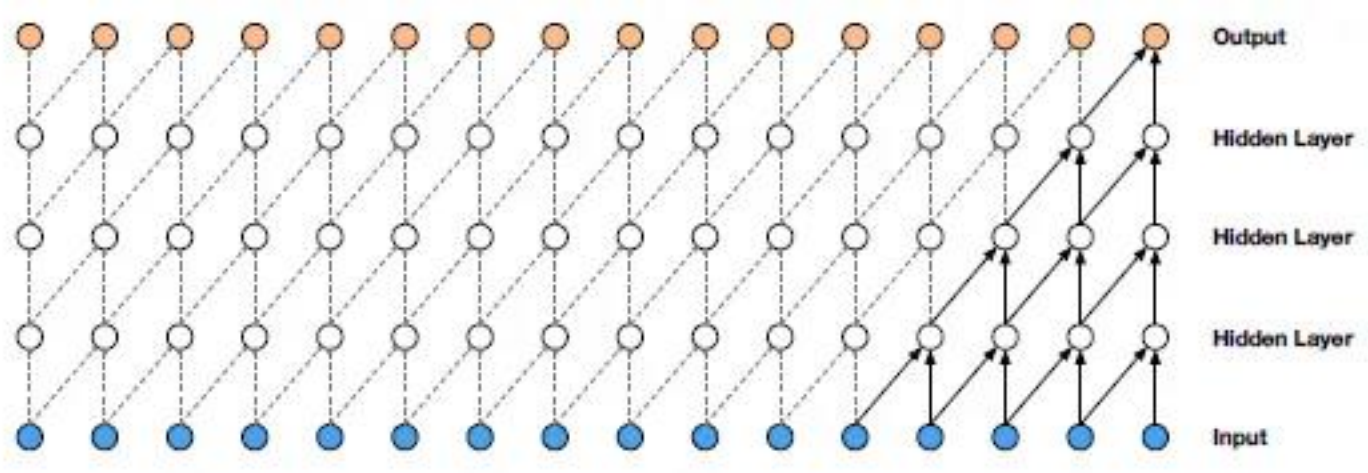
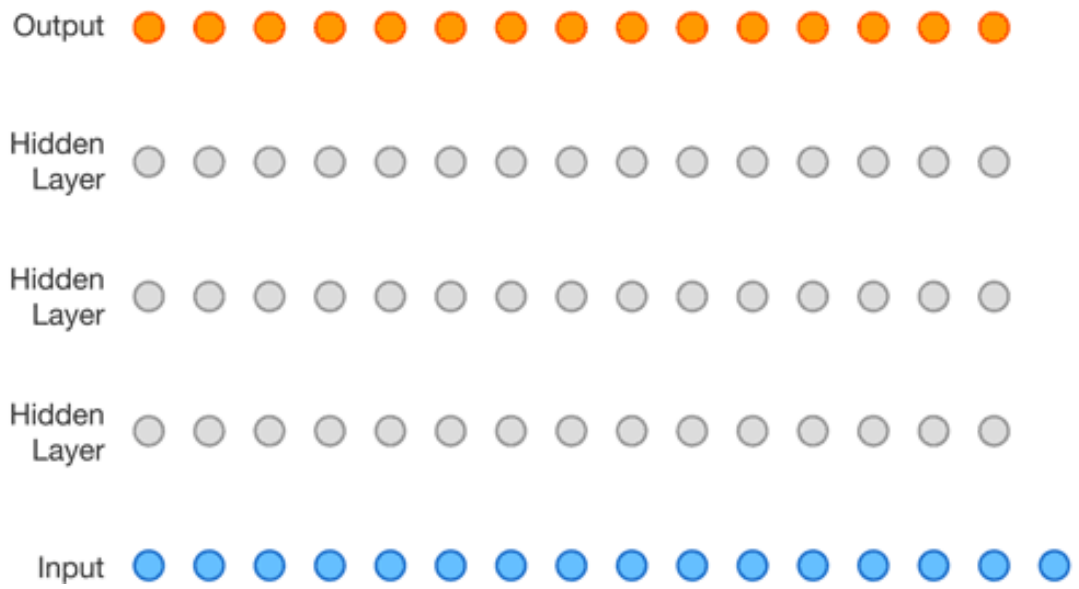



Figure 2: Visualization of a stack of causal convolutional layers.

# Fundamentação teórica

- Convoluções dilatadas: ampliam o campo receptivo de forma exponencial, permitindo que o modelo considere dependências de longo prazo no áudio



# Fundamentação teórica

- **Quantização μ-law**: aplica uma compressão não-linear que reduz o espaço para apenas 256 níveis discretos.
- Assim, o problema de prever a próxima amostra transforma-se em uma tarefa de classificação com 256 classes, resolvida por uma camada softmax.

$$f(x_t) = \text{sign}(x_t) \frac{\ln(1 + \mu|x_t|)}{\ln(1 + \mu)},$$

# Fundamentação teórica

- Unidades com portões: combinam funções *tanh* e *sigmoide*, permitindo controlar de forma não-linear quais informações passam adiante em cada camada.

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

- Conexões residuais e skip connections: permitem redes muito profundas. Essas conexões aceleram o treinamento e tornam possível empilhar dezenas de camadas de convoluções dilatadas.

# Fundamentação teórica

- O WaveNet pode ser condicional, ou seja, além do histórico de amostras, recebe uma entrada adicional $h$.
- Essa condição pode representar texto, identidade do falante ou qualquer outro sinal auxiliar, orientando a geração de áudio.

$$p(\mathbf{x} \mid \mathbf{h}) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1}, \mathbf{h}).$$

# Fundamentação teórica

Esse condicionamento pode ser implementado de duas formas:

- **Global**: uma única representação latente $h$, constante em toda a sequência.

Ex.: identidade do falante, idioma.

- **Local**: uma série temporal $ht(y)$, alinhada ao áudio.

Ex.: sequência de fonemas.

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

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

# Arquitetura e operação
# Estrutura do MobileNet v1

- A arquitetura do WaveNet começa com uma convolução causal inicial, que garante o respeito à ordem temporal.
- Em seguida, uma pilha de blocos residuais aplica convoluções dilatadas com portões (tanh × sigmoide).
- As conexões residuais e skip permitem treinar redes profundas e acumular informação de múltiplos níveis.
- Projeções 1×1 e uma softmax transformam os sinais acumulados em probabilidades sobre as próximas amostras da waveform.

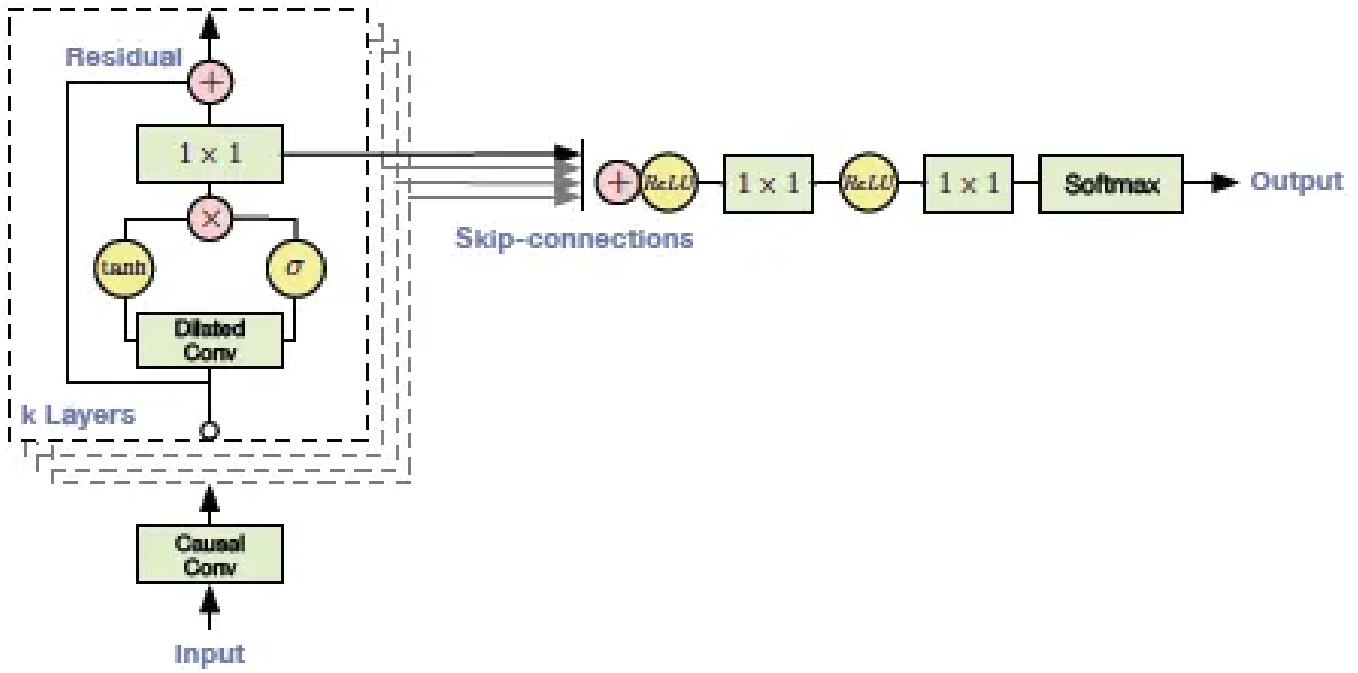



Figure 4: Overview of the residual block and the entire architecture.

# Treinamento e otimização

| | |
|---|---|
| Quantização u-Law | • Sinal de 16 bits reduzido a 256 valores discretos.<br>• Transforma o problema em classificação multiclasse. |
| Função de perda | • Uso de cross-entropy entre a saída prevista e a amostra real.<br>• Maximiza a probabilidade da classe correta. |
| Treinamento end-to-end | • Da waveform de entrada até a saída softmax.<br>• Ajuste de parâmetros via backpropagation. |
| Otimizador | • Adam, utilizado para atualização dos pesos.<br>• Garante convergência eficiente em redes profundas. |

# Treinamento e otimização

**Conexões residuais e skip**

- Permitem treinar redes muito profundas.
- Evitam problemas de desaparecimento de gradiente.

Batch Normalization

- Introduzida nos blocos convolucionais.
- Melhora a **estabilidade e velocidade** do treino.

Custo Computacional

- Gração extremadamente lenta em CPUs

# Vantagens

- ✔ Gera fala extremamente natural, também música e outros sons.
- ✔ Mais próximas de uma voz humana real (respirações, pausas e entonações variadas).
- ✔ Flexibilidade por modelar áudio bruto.
- ✔ Qualidade do áudio gerado e a versatilidade são as maiores vantagens do modelo WaveNet

# Desvantagens

Desempenho de inferência (síntese) extremamente lento.

Computacionalmente pesado também em memória.

Sem condicionamento de alto nível,  o WaveNet sozinho não garante coerência de larga escala

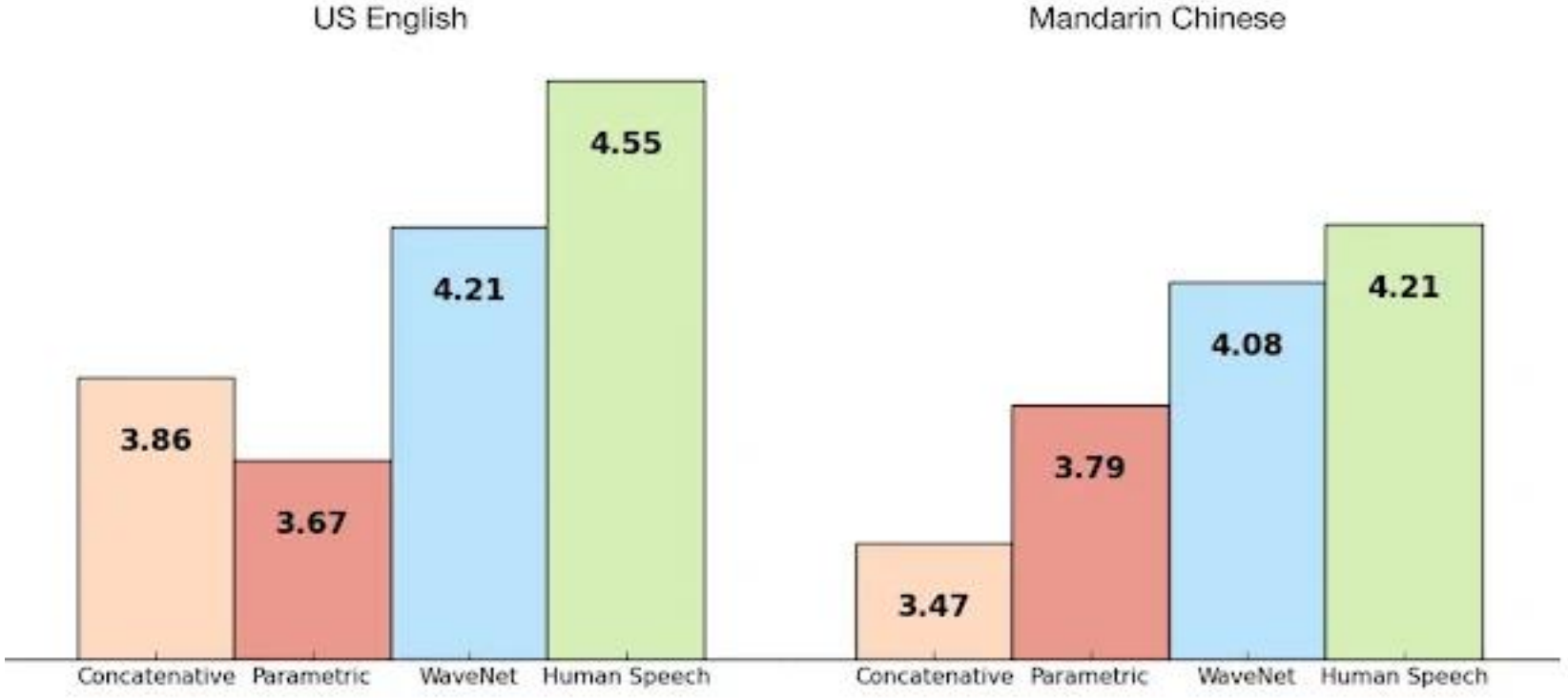
Treinamento exigente em dados, requer muitas horas de áudio e diversidade para generalizar.

# Exemplo(s) de aplicação

| Aplicação | Descrição | Resultados/Impacto |
|---|---|---|
| **TTS (Text-to-Speech)** | Conversão texto → fala natural | Google Assistant desde 2017 |
| **Música** | Geração de piano e outros estilos (condicionamento por tags) | Sons harmônicos, mas coerência global limitada |
| **Reconhecimento de fala** | Classificação de fonemas direto da waveform (TIMIT) | 18.8% PER – melhor resultado em áudio cru |
| **Conversão / Enhancing** | Conversão de voz, supressão de ruído, reconstrução de pacotes perdidos | Usado no Google Duo (WaveNetEQ) |

# Exemplo(s) de aplicação

- Histograma de qualidade (MOS, escala 1–5) obtido em testes cegos com ouvintes humanos. Comparamos três famílias de TTS, os sistemas paramétricos, sistemas concatenativos e o WaveNet, como também e incluímos a voz humana como referência superior.



# Exemplo(s) de aplicação

| | **US inglês** | **Mandarim Chinese** |
|---|---|---|
| **Parametric** | | |
| **Concatenative** | | |
| **WaveNet** | | |

# Exemplo(s) de aplicação

**Saber o que dizer**

Para usar o WaveNet para transformar texto em fala, temos que dizer a ele o que é o texto. Se treinarmos a rede sem a sequência de texto, ela ainda gera fala, mas agora ela tem que inventar o que dizer.

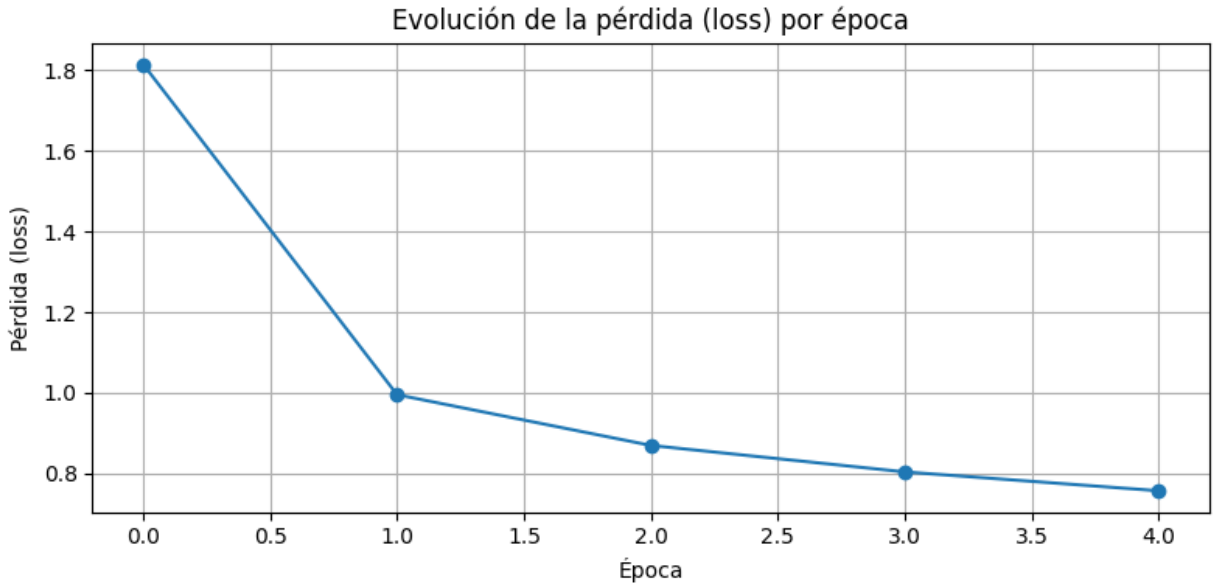
# Exemplo(s) de aplicação

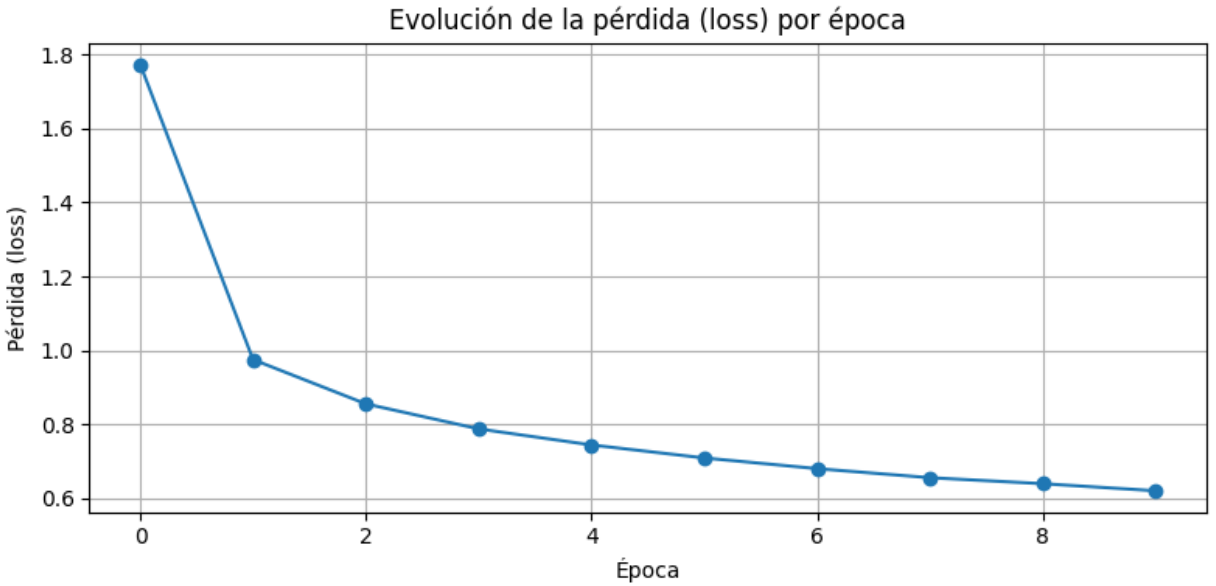
O WaveNet é capaz de aprender as características de muitas vozes diferentes. Alterando a identidade do locutor, podemos usar o WaveNet para dizer a mesma coisa em vozes diferentes:




# Exemplo(s) de aplicação

- Base de dados PhysioNet – Circor: 100 audios (5 segundos)
- Som Cardíaco Normal
- Filtro: 20 – 800 Hz
- 5 - 10 épocas

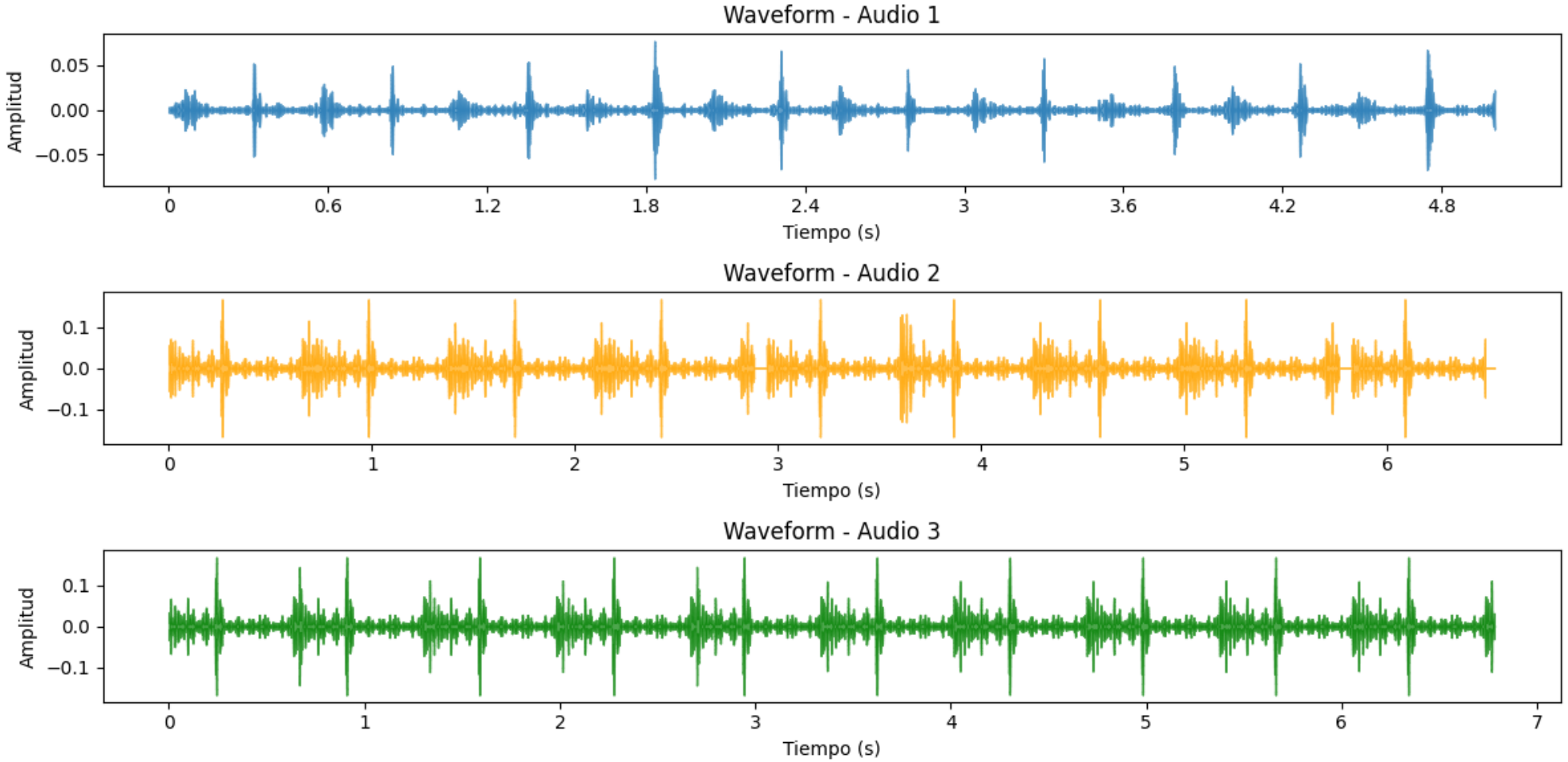


# Exemplo(s) de aplicação

**Geração**

- Sem épocas -> 1 áudio (5 segundos)
- 5 épocas ->  1 aúdio (~3 segundos*2) – 1h07m
- 10 épocas -> 1 aúdio (~3 segundos*2) – 1h32m



# Comparação com outros modelos

- A tabela mostra o WaveNet e seus sucessores, indicando como evoluíram em qualidade e velocidade. O WaveNet alcançou MOS 4,2, mas era lento; o Parallel WaveNet e o WaveGlow mantiveram qualidade similar com muito mais rapidez; já o DiffWave e o HiFi-GAN chegaram a uma qualidade próxima da voz humana em tempo real, tornando-se padrão nos TTS modernos.

| Modelo | Ano | Abordagem | Qualidade (MOS) | Velocidade | Usos típicos |
|---|---|---|---|---|---|
| **WaveNet** | 2016 | Convolucional (causal/dilatada) | 4.2 (inglês, mandarim) | Muito lento (amostra por amostra) | Geração incondicional |
| **Parallel WaveNet** | 2017 | Distilação do WaveNet | 4.41 | 1000× mais rápido | Google Assistant |
| **WaveGlow** | 2018 | Normalizing Flow + WaveNet | ≈ WaveNet | Inferência paralela, rápida | Text-To-Seepch (TTS) |
| **DiffWave** | 2020 | Modelo de difusão | 4.44 | Rápido ainda iterativo | TTS de alta fidelidade |
| **HiFi-GAN** | 2020 | GAN vocoder | Próximo voz humana | Tempo real | TTS em dispositivos, sistemas comerciais |

# Perguntas?

# Referências


- [1] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, “WaveNet: A generative model for raw audio,” *https://arxiv.org/abs/1609.03499*, 2016.
- [2] A. van den Oord, Y. Li, I. Babuschkin, et al., “Parallel WaveNet: Fast high-fidelity speech synthesis,” in *Proc. Int. Conf. on Machine Learning (ICML)*, 2018, pp. 3918–3926. http://proceedings.mlr.press/v80/oord18a.html
- [3] R. Prenger, R. Valle, and B. Catanzaro, “WaveGlow: A flow-based generative network for speech synthesis,” in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3617–3621. https://arxiv.org/abs/1811.00002
- [4] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, “DiffWave: A versatile diffusion model for audio synthesis,” *arXiv preprint arXiv:2009.09761*, 2020. https://arxiv.org/abs/2009.09761
- [5] J. Kong, J. Kim, and J. Bae, “HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 17022–17033. https://arxiv.org/abs/2010.05646

# Obrigado!

# Links

- Github: https://github.com/aadlrei/TP___558-Topicos-Avancados-em-Aprendizado-de-Maquina.git
- Quiz: https://forms.gle/LzTQVmbFg8976VQi8