Uma Revisão de Arquiteturas Ponta a Ponta para Sintetização de Voz

Lucy Anne de Omena Evangelista[1], **Patrícia do Nascimento**[1], **Carlos Eduardo Leão Elmadjian**[1], **Alfredo Goldman Vel Lejbman**[1]

[1]Instituto de Matemática e Estatística – Universidade de São Paulo (USP)

lucy.omena@usp.br, pathilink@gmail.com, elmad@ime.usp.br, gold@ime.usp.br

Abstract. *The objective of this work is to carry out a comparative bibliographic study between the architectures for voice synthesis (Char2Wav, ClariNet, Tacotron, Tacotron 2, and DeepVoice 3) by systematizing information regarding the resources and capacity of the architectures. The comparative study also covered the frameworks (TensorFlow, PyTorch, etc.) used in the implementation of the architectures. At the end, some informational points are suggested that should be taken as relevant when comparing the available architectures.*

Resumo. *O objetivo deste artigo é realizar um estudo bibliográfico comparativo entre as arquiteturas para sínteze de voz (Char2Wav, ClariNet, Tacotron, Tacotron 2 e DeepVoice 3), sistematizando informações quanto a recursos e capacidade das arquiteturas. O estudo comparativo também contemplou os frameworks (TensorFlow, PyTorch, etc.) utilizados na implementação das arquiteturas. Ao final, são sugeridos alguns pontos informacionais que devem ser tomados como relevantes ao se comparar as arquiteturas disponíveis.*

1. Introdução

Durante o processo de escolha de uma arquitetura para sintetização de voz, problemas como a falta de elementos objetivos como descrição, capacidade, desempenho e recursos necessários às arquiteturas impossibilitam que haja a realização de estudos comparativos entre as mesmas. Desta forma, o objetivo deste artigo é propor parâmetros de comparação entre arquiteturas com base em informações disponibilizadas de forma não padronizada nem sistematizada em diversos artigos.

Deste modo, realizamos uma revisão bibliográfica de diferentes arquiteturas ponta a ponta utilizadas para a sintetização de voz: Clarinet [Ping et al. 2018], Char2Wav [Sotelo et al. 2017], Tacotron [Wang et al. 2017], Tacotron 2 [Shen et al. 2018] e Deep Voice 3 [Ping et al. 2017]. Após a leitura dos artigos oficiais das arquiteturas, foram extraídas informações acerca da configuração, funcionamento e treinamento das mesmas. Com base nessas informações, informações externas e em configurações específicas selecionadas a partir de comprovada melhoria na sintetização de voz, foram selecionados critérios para seleção de arquiteturas.

2. Estudo das arquiteturas

A arquitetura Char2Wav [Sotelo et al. 2017], é uma das primeiras arquiteturas a utilizar a configuração reader-vocoder, e baseada em seq2seq (sequence-to-sequence). Gera

áudio a partir de texto em linguagem natural ou em fonemas. Seu Vocoder, estrutura para sintetização da voz a partir da saída da rede, é uma rede SampleRNN capaz de modelar dependências longas em dados sequenciais e perceber dinâmicas de modulação de acordo com a entrada fornecida.

Tacotron [Wang et al. 2017] é um modelo generativo com paradigma de atenção que sintetiza fala a partir de caracteres e, assim como a Char2Wav, se baseia em um modelo seq2seq. Requer pouco tratamento de dados e baseia-se em frames, tornando-o relativamente mais rápido do que modelos autorregressivos em nível de amostra. Utiliza o algoritmo Griffin-Lim como vocoder e possui um módulo CHBG capaz de reduzir o overfitting da rede e melhorar consideravelmente a qualidade de pronúncias.

O DeepVoice 3 [Ping et al. 2017] possui arquitetura seq2seq totalmente convolucional com mecanismo de atenção de posição aumentada. Permite o treinamento para síntese de voz de diversos locutores e permite treinamento rápido de diversos datasets a partir de processamento paralelo. Utiliza mecanismos de atenção em todas os seus módulos para evitar o uso de redes recorrentes e pode utilizar diversos vocoders. Necessita de uma grande quantidade de dados tratados para o treino de diversos locutores.

ClariNet [Ping et al. 2018] é a primeira arquitetura neural texto-para-onda (Text to Wave – TTW) feita para síntese de voz. Em seu artigo, é proposta uma configuração que permita a sintetização para múltiplos locutores que infere gênero e sotaque dos mesmos após o processo do Multi-Speaker ClariNet. Cada componente da rede recebe como viés de treinamento uma incorporação de locutor com poucas dimensões. Possui uma BridgeNet entre o Decoder e Encoder para melhoria da precisão.

Tacotron 2 [Shen et al. 2018] possui as mesmas características do Tacotron quanto a sua forma - redes recorrentes do tipo seq2seq. As melhorias do Tacotron 2 em relação ao seu anterior são: o mapeamento de agrupamentos de caracteres para espectrogramas Mel como entrada para o vocoder, que utiliza o modelo WaveNet. Assim, Tacotron 2 passa a ter uma estrutura totalmente baseada em redes neurais ponta a ponta.

Com o intuito de encontrar mais informações sobre as arquiteturas ponta a ponta sob análise foram pesquisadas páginas e repositórios. Nenhuma das arquiteturas sob estudo possuem implementações oficiais em repositórios abertos, com exceção da Char2Wav. Entretanto, todas possuem pelo menos uma implementação não oficial [1].

3. Metodologia

A figura 1 resume em uma tabela as informações levantadas. Sugerimos alguns critérios base para seleção de uma arquitetura de acordo com a finalidade da sintetização de voz e dos recursos disponíveis: tipo de vocoder utilizado; necessidade de pré-processamento dos dados; disposição da estrutura da rede nos módulos de atenção e saída da rede; valor da escala MOS (Mean Opinion Score); e existência de implementações e repositórios

[1]Char2Wav: *https://github.com/sotelo/parrot/blob/master/quantize.py*; ClariNet: *https://github.com/ksw0306/ClariNet*; Tacotron: *https://github.com/*{*keithito/tacotron,Kyubyong/tacotron,r9y9/tacotron_pytorch*} (TensorFlow, TensorFlow, e Pytorch); Tacotron 2: *https://github.com/*{*Rayhane-mamah/Tacotron-2, NVIDIA/tacotron2, nii-yamagishilab/tacotron2*} (TensorFlow, Pytorch, e TensorFlow com Spark); DeepVoice3: *https://github.com/*{*r9y9/deepvoice3_pytorch, Kyubyong/deepvoice3*} (Pytorch, TensorFlow).

	Clarinet Modificado [Ping et al. 2018]	Char2Wav [Sotelo et al. 2017]	Tacotron [Wang et al. 2017]	Tacotron 2 [Shen et al. 2018]	Deep Voice 3 [Ping et al. 2017]
Data de Lançamento	09/07/2019	15/04/2017	06/04/2017	16/02/2018	22/02/2018
Vocoder	Gaussian Autoregressive	SampleRNN	Griffin-Lim	WaveNet modificada	Griffin-Lim, WORLD & WaveNet
Pré-processamento é necessário	Sim.	-	Não.	Não.	Sim.
Normalização é necessária	Sim.	-	Sim.	Sim.	Sim.
Estrutura da rede	Encoder - Como em [Ping et al. 2017]. Decoder - Como em [Ping et al. 2017]. BridgeNet - Blocos convolucionais. Vocoder	Encoder - BRNN Decoder - Com atenção, RNN Vocoder	Encoder - PreNet & CBHG; Decoder - com atenção, Prenet & GRU RNN; Post Processing Net - CBHG Vocoder	Encoder - PreNet & bidirectional LSTM; Decoder - com atenção, Prenet & LSTM & camadas convolucionais; Post Processing Net - blocos convolucionais Vocoder	Encoder - com atenção, convolucional com speaker embedding Decoder - com atenção & PreNet, com speaker embedding, autoregressivo convolucional Converter - rede convolucional para pós processamento Vocoder
MOS (pontuação (vocoder) [Dataset - papel])	20 layers - 3.75 $\pm$ 0.42 30 layers - 3.9 $\pm$ 0.36 40 layers - 3.89 $\pm$ 0.28	-	3.82 $\pm$ 0.085 4.001 $\pm$ 0.087 [Shen et al. 2018] 2.07 $\pm$ 0.31 [VCTK][Ping et al. 2017] 3.78 $\pm$ 0.34 (WaveNet) [Ping et al. 2017]	4.526 $\pm$ 0.066 4.429 $\pm$ 0.071 4.148 $\pm$ 0.124 4.354 [Ping et al. 2017]	3.78 $\pm$ 0.3 [Interno] 3.44 $\pm$ 0.32 (WORLD) [VCTK] 3.01 $\mp$ 0.29 (Griffin-Lim)[VCTK]
Base de Dados	VCTK (inglês)	Dimex-100 (espanhol), VCTK (inglês), Blizzard (inglês), e Pavoque (alemão)	Interno, de inglês americano	Interno, de inglês americano	Interno, de inglês norte americano, VCTK (inglês) e LibriSpeech (inglês)
Hiperparâmetros disponíveis	Sim, no artigo.	Sim, no repositório.	Sim, no artigo.	Sim, no repositório.	Sim, no artigo.
Repositório aberto	Não.	Sim. https://github.com /sotelo/parrot	Sim. https://github.com/ keithito/ tacotron (não oficial)	Sim. https://github.com/ Rayhane-maman/ Tacotron-2 (não oficial)	Sim. https://github.com/ r9y9/deepvoice3_pytorch (não oficial)
Audio Samples disponibilizados	Sim. link1:https://multi-sp eaker-clarinet-demo.github.io	Sim. https://josesotelo .com/speechsynthesis	Sim. https://google.github .io/tacotron/publications/ tacotron/index.html	Sim. https://google.github .io/tacotron/publications/ tacotron2/	Sim. http://research.baidu. com/Blog/index-view?id=91
Ponto de atenção	Há speaker empedding em todas as partes da arquiteturas	-	Prediz espectrogramas lineares.	Prediz espectrogramas mel. As redes são treinadas separadamente	Não possui módulos recorrentes, em compensação há muitos módulos de atenção. Precisa de grande quantidade de dados, mas é mais rápido
Treinamento	Adam Optimizer com 1.5M passos, batch size de 16.	-	Adam Optimizer 2M passos, batch size de 32.	Adam Optimizer com taxa de aprendizagem de 10^{-4}, batch size de 64.	Converge após 500k passos.

Figura 1. Tabela comparativa das arquiteturas

abertos (oficiais ou não) que tenham caráter reprodutível. Os critérios foram selecionados pelos seguintes motivos.

A partir dos artigos pesquisados conclui-se que a utilização de espectrogramas Mel, ao invés de espectrogramas lineares, tornam a sintetização de voz mais eficiente [Ping et al. 2017][Shen et al. 2018] e que a utilização de WaveNet como vocoder melhora consideravelmente a naturalidade da voz gerada pelos módulos de atenção em sua arquitetura [Sotelo et al. 2017][Ping et al. 2017][Shen et al. 2018].

Em [Luong et al. 2015] é mostrado que a utilização de módulos de atenção melhora o aproveitamento de informações e na identificação de padrões. Estes mecanismos são utilizados em abundância nas arquiteturas mais atuais ([Shen et al. 2018][Ping et al. 2018][Ping et al. 2017]) sob a justificativa de aumentar a naturalidade da voz sintetizada, verificável a partir de *audio samples* disponibilizados.

A etapa de pré-processamento de dados, além de demandar tempo do desenvolvimento do modelo, pode afetar a qualidade da síntese por ser feita manualmente e, por conseguinte, sujeita a erros humanos. Arquiteturas que sejam capazes de lidar com o dado sem intervenções são as mais adequadas; e a escala MOS (naturalidade e compreensão) e a existência de implementações abertas objetivam a guiar a escolha de arquiteturas que possam ser balizadas na naturalidade percebida pelo áudio gerado e pelo caráter de reprodutibilidade e suporte à implementação da arquitetura para dados próprios. Vale ressaltar que a escala MOS é subjetiva e a sua comparação direta deve ser feita com cautela.

4. Discussão e conclusão

Com base nos critérios levantados, as arquiteturas Tacotron 2 e DeepVoice 3 são as mais adequadas para a síntese de áudio. Ambas as arquiteturas são eficientes no uso de módulos de atenção e são as mais inovadoras na área de sintetização de voz. Tacotron 2 é uma arquitetura de treinamento rápido e que consegue se especializar bem para um mesmo conjunto de dados, sendo mais adequada para o uso em pequena escala, enquanto a Deep Voice 3 possui maior aplicabilidade em escala comercial pela sua flexibilidade quanto aos locutores e vocoders, possuindo treinamento rápido e eficiente, mas que requer maior capacidade computacional e quantidade de dados.

As informações disponibilizadas acerca das necessidades computacionais e de dados para o treinamento dos modelos são escassas e não padronizadas. A construção da tabela comparativa auxiliou na comparação geral entre arquiteturas e com isso reafirmamos como a sistematização das informações contidas nos artigos é fundamental para o processo de escolha de uma arquitetura. Também acreditamos que informações como o volume de dados utilizados e seu perfil, quantidade de máquinas utilizadas, batch size e tempo médio de treino (ou ainda definição de iterações ou passos do modelo) são boas sugestões de informações que podem auxiliar no estudo comparativo das arquiteturas.

Neste sentido, as informações e metodologia levantadas neste artigo servem de base para futuros estudos e cumpre o seu objetivo de incentivar a padronização de informações disponibilizadas e maior transparência no desenvolvimento de modelos.

Agradecimentos

Agradecemos ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela concessão da bolsa de estudos de iniciação científica sob código 120151/2019-7. Este artigo faz parte da divisão de Interação Humano-Computador (IHC) do projeto A.D.A. (Assistente Distribuída Avançada), uma iniciativa do grupo *dev.research()* que faz parte do programa de extensão USP CodeLab. Agradecemos o grupo USP CodeLab pelo seu suporte, financiamento e incentivo.

Referências

Luong, M.-T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Ping, W., Peng, K., and Chen, J. (2018). Clarinet: Parallel wave generation in end-to-end text-to-speech. *arXiv preprint arXiv:1807.07281*.

Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., Raiman, J., and Miller, J. (2017). Deep voice 3: Scaling text-to-speech with convolutional sequence learning. *arXiv preprint arXiv:1710.07654*.

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., et al. (2018). Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4779–4783. IEEE.

Sotelo, J., Mehri, S., Kumar, K., Santos, J. F., Kastner, K., Courville, A., and Bengio, Y. (2017). Char2wav: End-to-end speech synthesis.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., et al. (2017). Tacotron: Towards end-to-end speech synthesis. *arXiv preprint arXiv:1703.10135*.