UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Síntese de fala aplicada à geração de conjunto de dados para reconhecimento automático de fala

Edresson Casanova

Tese de Doutorado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP
SÃO CARLOS

Edresson Casanova

Síntese de fala aplicada à geração de conjunto de dados para reconhecimento automático de fala

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Moacir Antonelli Ponti
Co-orientadora: Profa. Dra. Sandra Maria Aluísio

USP – São Carlos
Setembro de 2022

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

```
       Casanova, Edresson
C335s     Síntese de fala aplicada à geração de conjunto de
       dados para reconhecimento automático de fala /
       Edresson Casanova; orientador Moacir Antonelli
       Ponti; coorientadora Sandra Maria Aluísio. -- São
       Carlos, 2022.
          199 p.


          Tese (Doutorado - Programa de Pós-Graduação em
       Ciências de Computação e Matemática Computacional) --
        Instituto de Ciências Matemáticas e de Computação,
       Universidade de São Paulo, 2022.


          1. Síntese de fala. 2. Reconhecimento automático
       de fala. 3. Síntese de fala zero-shot. 4. Síntese de
       fala multilíngue. 5. Conversão de voz zero-shot. I.
       Antonelli Ponti, Moacir, orient. II. Aluísio, Sandra
       Maria, coorient. III. Título.
```

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

Edresson Casanova

Speech synthesis applied to the generation of datasets for automatic speech recognition

Thesis submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP – in accordance with the requirements of the Computer and Mathematical Sciences Graduate Program, for the degree of Doctor in Science. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Moacir Antonelli Ponti
Co-advisor: Profa. Dra. Sandra Maria Aluísio

USP – São Carlos
September 2022

AGRADECIMENTOS

Em primeiro lugar agradeço aos meus pais, à minha irmã e a toda minha família pelo amor e apoio nessa jornada de aprendizado.

A todos os incríveis professores que passaram pela minha vida, compartilhando seus conhecimentos e me transformando em quem sou hoje. Ao Prof. Dr. Hamilton Pereira da Silva da Universidade Tecnológica Federal do Paraná (UTFPR) por ter me convidado, no segundo ano da graduação, para fazer parte de um projeto de iniciação científica na área de robótica, e por todo o conhecimento compartilhado em aproximadamente um ano de orientação.

Ao Prof. Dr. Arnaldo Candido Junior da UTFPR, por ter aceitado me orientar em projetos de processamento de fala durante os últimos anos de graduação; pela incrível orientação durante meu Trabalho de Conclusão de Curso (TCC), e por ter me ensinado a escrever um texto científico. Agradeço por todo o conhecimento compartilhado, por ter me incentivado a realizar a aplicação para o processo seletivo de Doutorado Direto do Programa Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC) do ICMC-USP e pelo auxílio durante essa pesquisa, nas publicações em conjunto.

Aos professores Arnaldo Candido Junior, Hamilton Pereira da Silva e Pedro Luiz de Paula Filho, pela confiança no início dessa pesquisa com as cartas de recomendação para a pós-graduação.

Ao meu co-orientador não oficial Dr. Christopher Shulby pelo auxílio na procura de um orientador na área de processamento de fala, que no momento da aplicação para o doutorado não existia no ICMC-USP, bem como o apoio e discussões em reuniões semanais durante a realização dessa pesquisa.

Aos melhores orientadores que um estudante poderia almejar em uma pós-graduação, a Profa. Dra. Sandra Maria Aluísio e o Prof. Dr. Moacir Antonelli Ponti, pela confiança em terem aceitado me orientar de forma conjunta em uma área fora de suas respectivas áreas de atuação e por todo o conhecimento compartilhado em todo esse período. Se algum dia eu vier a orientar alguém, quero tentar ser pelo menos metade do que ambos foram para mim nesse período.

Ao Prof. Dr. Anderson da Silva Soares pela oportunidade de participar de projetos de pesquisa e desenvolvimento em processamento de fala no Centro de Excelência em Inteligência Artificial do estado de Goiás (CEIA), bem como todo o apoio durante a realização desta pesquisa.

Aos demais co-autores dos artigos publicados durante essa pesquisa. Se tivemos algum mérito no resultado, certamente foi em grande parte devido aos incríveis pesquisadores que

participaram dessa pesquisa. Em especial aos professores Anderson da Silva Soares, Frederico Santos de Oliveira, Marcelo Finger e Rogério Figueredo de Sousa; e aos pesquisadores Ana Carolina Rodrigues, Eren Gölge, Julian Weber, Marco Antonio Sobrevilla Cabezudo, Marcio Inácio, Marcos Treviso e Sidney Evaldo Leal.

Aos demais colegas e integrantes do Núcleo Interinstitucional de Linguística Computacional (NILC), pelas conversas, reuniões, festas, em geral, por todos os momentos de convivência que certamente foram fundamentais durante a realização dessa pesquisa.

Agradeço aos financiadores deste projeto: a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES); o Centro de Excelência em Inteligência Artificial do estado de Goiás (CEIA) e o *Center for Artificial Intelligence* (C4AI).

E, por fim, à USP e ao ICMC, que me aceitaram como aluno de Doutorado Direto e propiciaram um ambiente amigável e confortável durante o desenvolvimento dessa pesquisa.

“As invenções são, sobretudo,
o resultado de um trabalho de teimoso.”
(Alberto Santos Dumont)

RESUMO

CASANOVA, E. **Síntese de fala aplicada à geração de conjunto de dados para reconhecimento automático de fala**. 2022. 199 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

O reconhecimento automático de fala é um dos objetivos mais antigos da computação, pois reconhecer a fala oferece benefícios promissores para aplicações comerciais e pessoais. Ainda que os sistemas de reconhecimento automático de fala tenham evoluído com o advento de métodos *deep learning*, o reconhecimento automático de fala ainda não é um problema totalmente solucionado. Em muitos idiomas ainda há escassez de recursos livres, resultando em sistemas de reconhecimento automático de fala com baixo desempenho. Por outro lado, a área de síntese de fala também evoluiu na última década permitindo o surgimento de modelos de síntese de fala *zero-shot multi-speaker* que permitem gerar fala na voz de um locutor alvo utilizando apenas alguns segundos de fala desse locutor. Esses avanços motivaram o uso de síntese de fala *zero-shot multi-speaker* no treinamento de sistemas de reconhecimento automático de fala, com estudos mostrando que a síntese pode melhorar significativamente o desempenho de sistemas de reconhecimento automático de fala. Entretanto, os modelos de síntese *zero-shot multi-speaker* ainda necessitam de uma grande quantidade de locutores e horas de fala durante o treinamento, deste modo, inviabilizando a sua aplicação em idiomas com poucos recursos disponíveis. Nessa tese de doutorado, investigou-se o desenvolvimento e a avaliação de modelos de síntese de fala *zero-shot multi-speaker* em cenários com poucos locutores disponíveis. Para isso, propusemos o uso de modelos *flow-based*, devido ao seus resultados no estado da arte em síntese de fala. Além disso, investigou-se o uso de modelos multilíngues, deste modo, fazendo uso da quantidade de locutores disponíveis em idiomas com muitos recursos disponíveis. Os resultados alcançados com esse trabalho tornaram possível o desenvolvimento de sistemas de síntese de fala *zero-shot multi-speaker* e conversão de voz *zero-shot* em idiomas com poucos locutores disponíveis. Além disso, a abordagem proposta nesse trabalho foi aplicada na melhoria de sistemas de reconhecimento automático de fala em dois idiomas, simulando um cenário com apenas um locutor disponível para o treinamento do modelo síntese *zero-shot multi-speaker*. Apesar de utilizar apenas um locutor nos idiomas alvos, a abordagem de aumento de dados proposta nesse trabalho alcançou resultados comparáveis ao estado da arte no idioma Inglês. Por fim, foi explorado o treinamento de um modelo de reconhecimento automático de fala com um único locutor real nos idiomas alvos, utilizando a abordagem de aumento de dados proposta nesse trabalho, alcançando um *Word Error Rate* de 33.96% e 36.59%, respectivamente, para o conjunto de teste do dataset Common Voice no Português e no Russo.

Palavras-chave: Síntese de fala, reconhecimento automático de fala, síntese de fala *zero-shot*,

síntese de fala multilíngue, conversão de voz *zero-shot*.

ABSTRACT

CASANOVA, E. **Speech synthesis applied to the generation of datasets for automatic speech recognition**. 2022. 199 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

Automatic speech recognition is one the earliest goals of computing, as speech recognition offers promising benefits for business and personal applications. Although automatic speech recognition systems have evolved significantly with deep learning methods, it remains an open research problem. In many languages there is still a shortage of open/public resources, resulting in low-quality automatic speech recognition systems. On the other hand, speech synthesis has also evolved in the last decade, allowing for zero-shot multi-speaker TTS models to generate speech of a target speaker by using only a few seconds of its speech. These advances motivated the use of zero-shot multi-speaker TTS in the training of automatic speech recognition systems. Studies have shown that speech synthesis can significantly improve the performance of automatic speech recognition systems. However, the zero-shot multi-speaker TTS models still require a large number of diverse speakers and hours of speech during training, thus hindering their practical use in languages with less accessible data. In this work, we explored zero-shot multi-speaker TTS in scenarios with few available speakers. For this, we propose the use of flow-based models due to its state-of-the-art speech synthesis. Furthermore, we explored the use of multilingual models, seeking to leverage available data from languages with many available speakers. The results achieved by this work made possible the development of zero-shot multi-speaker TTS and zero-shot voice conversion systems in languages with few available speakers. Furthermore, the approach proposed in this work was applied to improve automatic speech recognition systems in two languages, simulating a scenario with only one speaker available for the training of the zero-shot multi-speaker TTS model. Despite using only one speaker in the target languages, our data augmentation approach achieved results comparable to the state-of-the-art in the English language. In addition, we explored the training of an automatic speech recognition model with a single real speaker in the target languages, using our data augmentation approach, reaching a Word Error Rate of 33.96% and 36.59%, respectively, for the test set of the Common Voice dataset in Portuguese and Russian.

Keywords: TTS, speech synthesis, ASR, automatic speech recognition, zero-shot multi-speaker TTS, multi-lingual speech synthesis, cross-lingual zero-shot voice conversion.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ANN	*Artificial Neural Network*
ASR	*Automatic Speech Recognition*
CER	*Character Error Rate*
CNNs	*Convolutional Neural Networks*
CTC	*Connectionist Temporal Classification*
DCNNs	*Deep Convolutional Neural Networks*
DCT	*Discrete Cosine Transform*
DCTTS	*Deep Convolutional Text to Speech*
DNNs	*Deep Neural Networks*
DRNN	*Deep Recurrent Neural Networks*
EER	*Equal Error Rate*
ELBO	*Evidence Lower Bound*
F0	*Fundamental Frequency*
FT	*Fourier Transform*
GANs	*Generative Adversarial Networks*
GE2E	*Generalized End-To-End*
GRU	*Gated Recurrent Unit*
GST	*Global Style Tokens*
KL	*Kullback-Leibler*
LAS	*Listen, Attend and Spell*
LDE	*Learnable Dictionary Encoding*
LSTM	*Long Short-Term Memory*
MFCCs	*Mel-Frequency Cepstral Coefficients*
MLP	*Multilayer Perceptron*
MOS	*Mean Opinion Score*
MOS	*Mean Opinion Score*
MSE	*Mean Squared Error*
NF	*Normalizing Flows*
ReLU	*Rectifier Linear Unit*
RNNs	*Recurrent Neural Networks*
SECS	*Speaker Encoder Cosine Similarity*

SECS	*Speaker Encoder Cosine Similarity*
Sim-MOS	*Similarity Mean Opinion Score*
SOTA	*State-Of-The-Art*
TTS	*Text to Speech*
VAE	*Variational Autoencoder*
WER	*Word Error Rate*

SUMÁRIO

CAPÍTULO

1

INTRODUÇÃO

1.1 Contexto e Revisão da Literatura

A fala é um meio essencial e natural de comunicação entre pessoas. Estudar esse processo em sistemas computacionais de forma a permitir a comunicação entre máquina e humano é de grande importância para o desenvolvimento de novas tecnologias. Estudos na área de Inteligência Artificial vêm se tornando fundamentais para a construção de métodos que tentam aprender conceitos, sendo o *deep learning* (ou aprendizagem profunda, também chamado de aprendizado profundo) (GOODFELLOW *et al.*, 2016) uma das abordagens mais proeminentes com esse fim e que se utiliza principalmente de redes neurais artificiais com múltiplas camadas de processamento (PONTI *et al.*, 2017). Nesse contexto, pesquisas relacionadas ao reconhecimento automático de fala (HANNUN *et al.*, 2014; AMODEI *et al.*, 2016; CHAN *et al.*, 2016; QUINTANILHA; BISCAINHO; NETTO, 2017; SCHNEIDER *et al.*, 2019; LI *et al.*, 2019a; BAEVSKI *et al.*, 2020; GULATI *et al.*, 2020; MAJUMDAR *et al.*, 2021; HSU *et al.*, 2021; BAEVSKI *et al.*, 2022) e síntese de fala (KYLE; JOSE; SOTELO, 2017; WANG *et al.*, 2017; SHEN *et al.*, 2018; TACHIBANA; UENOYAMA; AIHARA, 2018; PING *et al.*, 2018; KIM *et al.*, 2020; VALLE *et al.*, 2020) foram exploradas nas últimas décadas em conjunto com avanços na área de Inteligência Artificial.

Uma das aplicações dos sistemas de *Automatic Speech Recognition* (ASR), em conjunto com os sistemas de síntese de fala, é transformar a forma como o ser humano interage com a máquina, fazendo essa interação ser a mais próxima de uma conversa. O ASR pode ser visto como um dos objetivos mais antigos da computação (YU; DENG, 2016; BENESTY; SONDHI; HUANG, 2007), pois reconhecer a fala oferece benefícios promissores para aplicações comerciais e pessoais. Por exemplo, as empresas podem vender produtos mais acessíveis para seus clientes (e.g. legendadores automáticos (RAMANI *et al.*, 2020)), cidadãos em geral podem utilizar a tecnologia em qualquer lugar/momento para interesses pessoais (e.g. pesquisar sem a necessidade de digitar a questão de busca (TULSHAN; DHAGE, 2018)), e deficientes auditivos podem ter maior acessibilidade digital (HOLTER *et al.*, 2000). De fato, modelos de ASR estão atualmente disponíveis comercialmente na forma de assistentes virtuais inteligentes como Amazon Alexa (PURINGTON *et al.*, 2017), Google Home (DEMPSEY, 2017) e Apple Siri (GRUBER, 2009), cujas interações tendem a facilitar as tarefas diárias.

Os primeiros sistemas de ASR focavam no reconhecimento de números e não de palavras (ABOUELHASAN *et al.*, 2020). Um dos primeiros reconhecedores automático de fala, chamado Audrey, foi desenvolvido nos Laboratórios Bell em 1952 (MALIK *et al.*, 2021; ABOUELHASAN *et al.*, 2020). Audrey podia distinguir entre diferentes dígitos falados por um único usuário (DAVIS; BIDDULPH; BALASHEK, 1952). Na década de 1970, pesquisadores desenvolveram um sistema que podia reconhecer 203 palavras (VELICHKO; ZAGORUYKO, 1970). Durante essa década pesquisadores exploraram programação dinâmica (SAKOE; CHIBA, 1978) e algoritmos de reconhecimento de padrões (VELICHKO; ZAGORUYKO, 1970) na tarefa de ASR (ABOUELHASAN *et al.*, 2020). No início da década de 1980, os Modelos Ocultos de Markov (do inglês, *Hidden Markov Models* — HMMs), foram aplicados em ASR (ABOUELHASAN *et al.*, 2020). Desde então, HMMs continuaram a ser explorados e aprimorados ao longo dos anos, melhorando o desempenho dos sistemas de ASR gradualmente (GALES; YOUNG *et al.*, 2008). A partir de 1988, duas agências governamentais dos EUA, o *Defense Advanced Research Project Agency* (DARPA) e o *National Institute of Standards and Technology* (NIST), iniciaram avaliações mundiais para o reconhecimento de fala contínua (BAKER *et al.*, 2009), possibilitando um grande avanço na área de ASR. Os efeitos desse avanço podem ser vistos no crescimento do vocabulário de reconhecimento dos sistemas de ASR, o qual subiu de 900 palavras (1988-1992) (PAUL; BAKER, 1992) para 20.000 palavras (1993-1995) (BAKER *et al.*, 2009). Atualmente, é desejável que o vocabulário de sistemas de ASR tenha a maior cobertura possível do vocabulário do idioma que se quer reconhecer (CHAN *et al.*, 2016).

Os sistemas tradicionais de ASR são tipicamente compostos de muitos módulos específicos: um *front-end*, um dicionário fonético, um modelo acústico, um modelo de língua e um decodificador (GALES; YOUNG *et al.*, 2008), os quais exigem muitas escolhas que tornam difícil seu desenvolvimento. Em contrapartida, o aprendizado profundo tem o potencial de unir todos os módulos de um sistema ASR tradicional em um único modelo, tipicamente utilizando redes neurais artificiais, conectando diretamente a entrada falada à saída textual. Esse tipo de técnica é chamada de aprendizagem de ponta a ponta (*end-to-end learning*, em inglês), e inclui comumente a otimização de uma função objetivo que compara a saída obtida com a saída esperada. Tal técnica possui a desvantagem de possuir difícil interpretação, visto que os diversos passos bem definidos dos sistemas tradicionais são substituídos por unidades de processamento menos explícitas. No entanto, os sistemas de ASR treinados de ponta a ponta alcançaram desempenhos superiores, como por exemplo: DeepSpeech (HANNUN *et al.*, 2014), DeepSpeech 2 (AMODEI *et al.*, 2016), *Listen, Attend and Spell* (LAS) (CHAN *et al.*, 2016), Wav2letter (COLLOBERT; PUHRSCH; SYNNAEVE, 2016), Jasper (LI *et al.*, 2019a), Wav2vec (SCHNEIDER *et al.*, 2019), Wav2vec 2.0 (BAEVSKI *et al.*, 2020), Conformer (GULATI *et al.*, 2020), Citrinet (MAJUMDAR *et al.*, 2021), Hubert (HSU *et al.*, 2021) e Data2vec (BAEVSKI *et al.*, 2022), que estimam diretamente o texto a partir de um clipe de áudio em formato wav[1] ou uma representação espectral de entrada. A função objetivo (ou de custo, no caso de problemas de otimização relacionados à minimização) dos métodos DeepSpeech, DeepSpeech 2 e Jasper é chamada de *Connectionist Temporal Classification* (CTC) (GRAVES *et al.*, 2006; GRAVES; MOHAMED; HINTON, 2013). Mais recentemente, ao invés de empregar a CTC, o uso de unidades recorrentes e mecanismos de atenção (BAHDANAU; CHO; BENGIO, 2015; GEHRING *et al.*, 2017) em métodos como o LAS obtiveram destaque com resultados promissores (CHAN *et al.*, 2016; BAHDANAU *et al.*, 2016; CHIU *et al.*, 2018). Apesar disso, modelos estado da arte, como Wav2vec 2.0 (BAEVSKI *et al.*, 2020) e Hubert (HSU *et al.*, 2021), ainda utilizam a função CTC durante o treinamento para a tarefa ASR.

Por outro lado, os sistemas de síntese de fala, também conhecidos como *Text to Speech* (TTS), que têm como objetivo transformar um trecho de texto em fala natural e inteligível, avançaram bastante nos últimos anos. Assim como em ASR, sistemas tradicionais de síntese de fala também requerem a integração elaborada de muitos módulos específicos (TACHIBANA; UENOYAMA; AIHARA, 2018). Um sistema de síntese de fala tradicional inclui um analisador de texto, conversor grafema-para-fonema, estimador de duração, gerador de *Fundamental Frequency* (F0), gerador de espectro e codificador de voz (*vocoder*, em inglês). Dado um texto de entrada, o módulo analisador de texto converte datas, símbolos de moeda, abreviações, acrônimos e números em seus formatos padrão para serem pronunciados ou lidos pelo sistema, ou seja, realiza a normalização do texto e aborda problemas como homógrafos. Depois, com o texto normalizado, o analisador fonético converte grafemas em fonemas. Por sua vez, a duração de cada fonema é estimada. Então, o modelo acústico é usado para gerar características acústicas como F0 e espectral que corresponde às características linguísticas. Finalmente, o *vocoder* converte o espectro em uma forma de onda (ZE; SENIOR; SCHUSTER, 2013).

Novamente, o aprendizado profundo pode ser utilizado para integrar esses módulos em um único modelo ponta a ponta. A partir de 2010, a síntese de fala baseada em redes neurais artificiais (KYLE; JOSE; SOTELO, 2017; WANG *et al.*, 2017; SHEN *et al.*, 2018; TACHIBANA; UENOYAMA; AIHARA, 2018; ARIK *et al.*, 2017; GIBIANSKY *et al.*, 2017; PING *et al.*, 2018; LI *et al.*, 2019b; BIŃKOWSKI *et al.*, 2019; KIM *et al.*, 2020; VALLE *et al.*, 2020; DONAHUE *et al.*, 2021) gradualmente se tornou o método dominante e alcançou uma qualidade de fala muito mais natural (TAN *et al.*, 2021). Os modelos de síntese de fala baseados no aprendizado profundo geralmente são compostos por dois módulos, um modelo acústico que converte uma representação textual em uma representação intermediária da fala (e.g., espectrogramas de mel) e um *vocoder* que converte essa representação para áudio (formato wav); esses sistemas são conhecidos como sistemas de dois estágios (KIM; KONG; SON, 2021; TAN *et al.*, 2021).

Devido à característica sequencial dos dados de texto e áudio, unidades recorrentes

[1] Formato-padrão de arquivo de áudio para armazenamento digital.

foram os blocos de construção escolhidos para as redes neurais nos modelos Tacotron 1 e 2 (WANG *et al.*, 2017; SHEN *et al.*, 2018). Por sua vez, camadas convolucionais mostraram reduzir os custos, conforme apresentado nos modelos DeepVoice 3 (PING *et al.*, 2018) e *Deep Convolutional Text to Speech* (DCTTS) (TACHIBANA; UENOYAMA; AIHARA, 2018). Com a recente popularização de arquiteturas baseadas em *Transformers* (VASWANI *et al.*, 2017), alguns modelos de síntese de fala baseados nessa arquitetura surgiram, como Li *et al.* (2019b), que alcançou um desempenho semelhante ao Tacotron 2 (SHEN *et al.*, 2018), porém com treinamento 4,25 vezes mais rápido. Recentemente, os modelos baseados em fluxo (*flow-based models*, em inglês) (KINGMA *et al.*, 2016; HOOGEBOOM; BERG; WELLING, 2019; DURKAN *et al.*, 2019) receberam atenção nessa área, com destaque para o Flowtron (VALLE *et al.*, 2020) que alcançou resultados comparáveis ao Tacotron 2, permitindo alterar características prosódicas como velocidade da fala, dentre outras. Paralelamente, Kim *et al.* (2020) propuseram o GlowTTS, que alcançou um desempenho semelhante ao do Tacotron 2, sintetizando fala 15,7 vezes mais rápido. Por fim, devido os sistemas de dois estágios necessitarem de treinamento sequencial ou ajuste fino (onde o vocoder é ajustado utilizando espectrogramas extraídos do modelo de síntese de fala) para gerar fala de alta qualidade, recentemente os modelos EATS (DONAHUE *et al.*, 2021), FastSpeech 2s (REN *et al.*, 2020), EfficientTTS (MIAO *et al.*, 2021) e VITS (KIM; KONG; SON, 2021) integraram o modelo acústico com o *vocoder*, mostrando a viabilidade de sintetizar fala de alta qualidade com modelos de síntese de fala treinados de ponta a ponta. Para tornar essa difícil tarefa possível, Ren *et al.* (2020) e Donahue *et al.* (2021) utilizaram espectrogramas de mel como representação intermediária para auxiliar no aprendizado da representação do texto, deste modo, utilizando a representação intermediária predita pelo modelo acústico como entrada para o *vocoder* (KIM; KONG; SON, 2021). Apesar de alcançar resultados promissores, Ren *et al.* (2020) e Donahue *et al.* (2021) reportaram uma qualidade de fala ligeiramente inferior a de sistemas de dois estágios. Por outro lado, Kim, Kong e Son (2021) propuseram o modelo VITS, que não utiliza nenhuma representação intermediária da fala. Os autores propuseram o uso de um *Variational Autoencoder* (VAE) (KINGMA; WELLING, 2013) para conectar os dois módulos do sistema de síntese de fala por meio de variáveis latentes, permitindo assim um aprendizado de ponta a ponta eficiente e evitando limitações do uso de representações intermediárias (KIM; KONG; SON, 2021). Deste modo, o modelo VITS não depende de uma representação intermediária pré-computada (e.g, espectrograma mel), permitindo que o modelo aprenda uma representação próxima da ideal. Os autores mostraram que o sistema proposto supera os sistemas de dois estágios e alcança um resultado muito próximo da qualidade da fala humana.

Os avanços na área da síntese de fala motivaram pesquisas cujo objetivo é sintetizar a fala de um locutor alvo utilizando apenas alguns segundos de fala desse locutor. Essa abordagem é conhecida como *zero-shot multi-speaker TTS*, explorada inicialmente por Arik *et al.* (2018). Os autores estenderam a topologia do modelo DeepVoice 3 (PING *et al.*, 2018) e adicionaram *embeddings*[2] de locutores extraídos de um sistema externo de verificação de locutores. Para realizar tarefas *zero-shot* é comum a complementação das representações por meio da adição de informação externa contribuindo para enriquecer os dados, permitindo trabalhar em cenários de baixa amostragem e tornando a representação mais robusta a ruído nos dados de entrada (RESENDE; PONTI, 2022). Por outro lado, Jia *et al.* (2018) exploraram o Tacotron 2 (SHEN *et al.*, 2018), usando *embeddings* externos extraídos de um sistema de verificação de locutores treinado utilizando a função de perda *Generalized End-To-End* (GE2E) (WAN *et al.*, 2018). Como resultado, o modelo proposto é capaz de gerar fala semelhante a do locutor-alvo, mas não o suficiente para ser confundida com o locutor real. Seguindo o mesmo caminho, Cooper *et al.* (2020) explorou o Tacotron 2 com diferentes sistemas de verificação de locutores para a extração de *embeddings* de locutores utilizando *Learnable Dictionary Encoding* (LDE) (CAI; CHEN; LI, 2018), atingindo maior similaridade e fala natural para locutores não vistos durante o treinamento. No entanto, os autores mostraram que ainda existe uma diferença na similaridade da fala de locutores vistos e não vistos no treinamento. Para diminuir essa lacuna, Paul, Pantazis e Stylianou (2020) propuseram uma modificação no *vocoder* WaveRNN (KALCHBRENNER *et al.*, 2018), denominada SC-WaveRNN, que recebe, além do espectrograma mel, um *embedding* do locutor extraído de um sistema de verificação de locutores pré-treinado. Os autores mostraram que com o uso do SC-WaveRNN o sistema atinge uma maior semelhança para locutores não vistos, desta forma diminuindo a lacuna. Adicionalmente, Choi *et al.* (2020) propuseram o Attentron, que consiste em um codificador de granularidade fina, com mecanismo de atenção para extrair estilos detalhados de várias amostras de referência e um codificador de granularidade grossa, que extrai informações gerais da fala e ajuda a estabilizar a saída. Com o uso de várias amostras de referência, ao invés de apenas uma, o modelo alcança uma melhor semelhança para falantes vistos e não vistos no treinamento. Por fim, Kumar *et al.* (2021) explorou uma arquitetura baseada em *Transformers*. Os autores propuseram uma *normalization architecture* e um sistema de verificação de locutores baseado no Wav2vec 2.0 (BAEVSKI *et al.*, 2020). Os autores condicionaram a *normalization architecture* com *embeddings* de locutor, *pitch* e *energy*, aplicando a *normalization architecture* no *encoder* e no *decoder* do modelo. Apesar de resultados promissores, os autores não compararam o modelo proposto com nenhum dos trabalhos relacionados supracitados.

Os avanços em síntese de fala em geral e síntese de fala *multi-speaker zero-shot* também motivaram trabalhos como os de Li *et al.* (2018), Rosenberg *et al.* (2019) e Laptev *et al.* (2020), os quais mostraram que um sistema *zero-shot multi-speaker TTS* ou *multi-speaker* com manipulação de fala pode ser aplicado para aumento de dados de treinamento, com significativa melhoria de desempenho de modelos ASR na língua inglesa.

[2] Características numéricas obtidas por meio da transformação dos dados originais, que são comumente aprendidas por meio de uma rede neural artificial, de forma a ser efetiva para uma dada tarefa. Poderia ser traduzido para o Português como “imersão”, mas é pouco utilizado nessa forma na literatura e por isso adotamos o termo em Inglês.

Por fim, os recentes avanços em síntese de fala também motivaram pesquisadores a projetar modelos que podem aprender mais de um idioma ao mesmo tempo; Cao *et al.* (2019), Zhang *et al.* (2019), Nekvinda e Dušek (2020), Li *et al.* (2021) são exemplos dessa abordagem. Alguns desses modelos são particularmente interessantes, pois permitem a realização de *code-switching*, i.e. alterar o idioma durante a pronúncia de uma oração, deste modo, sendo possível pronunciar estrangeirismos de maneira eficiente na língua alvo (NEKVINDA; DUŠEK, 2020). Essa característica é relevante para essa pesquisa de doutorado, dado que o *code-switching* pode permitir o uso de locutores de um idioma em outro idioma.

1.2 Lacunas

Apesar do avanço alcançado com métodos baseados em *deep learning*, a tarefa de ASR não é um problema totalmente solucionado (DENG; LI, 2013; PETKAR, 2016). Enquanto sistemas de ASR em inglês são particularmente avançados devido à vasta disponibilidade de dados e recursos, em outras línguas como o português os resultados são significativamente inferiores devido ao volume e qualidade dos dados públicos (QUINTANILHA; NETTO; BISCAINHO, 2020).

Apesar de trabalhos como os de Li *et al.* (2018), Rosenberg *et al.* (2019) e Laptev *et al.* (2020) mostrarem que modelos *zero-shot multi-speaker TTS* podem ser empregados no aumento de dados para a tarefa de ASR e melhorarem o desempenho dos sistemas no idioma inglês, os modelos *zero-shot multi-speaker TTS* ainda requerem conjuntos de dados de alta qualidade e uma grande quantidade de locutores e horas de fala para convergirem. Geralmente, os modelos *zero-shot multi-speaker TTS* são treinados na língua inglesa com os conjuntos de dados VCTK (VEAUX *et al.*, 2016) e LibriTTS (ZEN *et al.*, 2019). O VCTK é um conjunto de dados que consiste em 44 horas de fala de 109 falantes. Por outro lado, o LibriTTS consiste em 586 horas de 2.456 falantes. No entanto, a grande maioria dos idiomas não têm um conjunto de dados disponível publicamente para treinamento de modelos de síntese de fala e os poucos conjuntos disponíveis são geralmente compostos por poucos locutores.

1.3 Questão de Pesquisa e Hipótese

Dado que os sistemas *zero-shot multi-speaker TTS* exigem *datasets* com um grande número de locutores para sua convergência, é possível superar essa limitação e obter um sistema *zero-shot multi-speaker TTS* em idiomas para os quais o número de locutores disponíveis tende a um?

A hipótese é a de que um modelo *flow-based*, como o GlowTTS, adaptado para treinamento *zero-shot multi-speaker* pode alcançar a convergência com uma menor quantidade de locutores. Também, que é possível realizar um treinamento usufruindo da quantidade de locutores presentes em outros idiomas e, desta forma, diminuir a quantidade necessária de locutores para treinamento no idioma alvo.

1.4 Objetivos

1.4.1 Objetivo Geral

Investigar uma abordagem para o desenvolvimento de um método *zero-shot multi-speaker TTS* para idiomas com *datasets* com pequena quantidade de locutores disponíveis. Deste modo, tornando mais viável a utilização de síntese de fala aplicada à tarefa de ASR em idiomas com poucos recursos disponíveis. Além da baixa disponibilidade de locutores, o comportamento desses métodos em idiomas diferentes do inglês, em particular o português, e a investigação de métodos que atuem com múltiplos idiomas também estão no escopo desse trabalho.

1.4.2 Objetivos Específicos

Para atingir o objetivo geral foram definidos os seguintes objetivos específicos:

1. Desenvolver e disponibilizar publicamente um *dataset* para síntese de fala em português brasileiro;

2. Propor um novo modelo *zero-shot multi-speaker TTS* que melhore a similaridade da fala, calculada via Similarity *Mean Opinion Score* (MOS) ou *Speaker Encoder Cosine Similarity* (SECS) (apresentadas na Seção 2.13), para locutores não vistos no treinamento e que alcance bons resultados com uma menor quantidade de locutores;

3. Investigar e propor adaptações ao modelo proposto em (2) para o treinamento com vários idiomas;

4. Explorar a aplicação do modelo proposto em (3) no treinamento de modelos de ASR e disponibilizar publicamente um grande *dataset* de fala espontânea, principalmente, e preparada do português brasileiro.

1.5 Organização

Esta monografia utiliza o modelo de Coleção de Artigos, trazendo 10 artigos publicados e/ou submetidos, e está dividida em 9 capítulos. A apresentação dos artigos não segue a ordem cronológica de sua publicação; os mesmos foram organizados por tema principal de modo a fornecer uma sequência natural aos assuntos, facilitando a leitura.

O Capítulo 2 apresenta os fundamentos das áreas de Inteligência Artificial e Processamento de Fala. Além disso, o capítulo também apresenta trabalhos relacionados à pesquisa de doutorado que não foram citados nos artigos dessa tese. O Capítulo 3 detalha conceitos e apresenta uma revisão da literatura sobre a aplicação de *deep learning* nas áreas de Síntese de Fala e Verificação de Locutores. O Capítulo 4 traz o artigo que contribui com a questão da falta de dados para síntese de fala no português brasileiro. O artigo apresenta o *TTS-Portuguese Corpus* que é o primeiro *dataset* disponível publicamente para treinamento de modelos profundos de síntese de fala em português brasileiro. O *dataset* é composto por aproximadamente 10,5 horas de fala de um único locutor. O Capítulo 5 apresenta um novo modelo *zero-shot multi-speaker TTS*, denominado *SC-GlowTTS*, que alcança uma maior similaridade para locutores não vistos no treinamento. Adicionalmente, o modelo consegue alcançar resultados competitivos com o treinamento em um *dataset* constituído por apenas 11 locutores. Além disso, o *SC-GlowTTS* sintetiza fala de alta qualidade na voz de um locutor alvo, utilizando apenas alguns segundos de fala desse locutor em tempo real na CPU. O Capítulo 6 apresenta o modelo YourTTS. YourTTS é um modelo *zero-shot multi-speaker TTS* multilíngue baseado nos modelos VITS e SC-GlowTTS, que alcança resultados no estado da arte em *zero-shot multi-speaker TTS* e resultados comparados ao estado da arte em *zero-shot voice conversion* no idioma inglês. Além disso, o modelo alcançou resultados promissores em *zero-shot voice conversion* e *zero-shot multi-speaker TTS* utilizando apenas um único locutor em um idioma alvo. Por fim, para locutores que possuem características de fala muito diferentes das vistas no treinamento o modelo YourTTS pode ser ajustado utilizando apenas 1 minuto de fala desses locutores e alcançar resultados estado da arte. O Capítulo 7 apresenta a aplicação do modelo YourTTS no aumento de dados para o treinamento de modelos ASR. Foram realizados experimentos explorando apenas um locutor no treinamento do modelo YourTTS nos idiomas alvos e, apesar disso, o método de aumento de dados proposto melhorou o desempenho do modelo ASR de forma comparável ao estado da arte no idioma inglês. Deste modo, mostrou-se que é possível utilizar síntese de fala para aumento de dados em um idioma alvo, mesmo que se tenha apenas um locutor nesse idioma. Além disso, o método proposto alcançou resultados promissores no treinamento de um modelo ASR utilizando apenas um locutor real no idioma alvo, abrindo novas possibilidades para o treinamento de modelos ASR em idiomas com poucos recursos disponíveis. O Capítulo 8 apresenta um grande *dataset* para ASR no português brasileiro, composto por 290,77 horas de fala espontânea e preparada, bem como disponibiliza publicamente um modelo ASR treinado nesse dataset.

Por fim, o Capítulo 9 apresenta um resumo das contribuições dessa pesquisa para a área de Processamento de Fala, trabalhos futuros e uma lista com todas as publicações realizadas no decorrer dessa pesquisa. Adicionalmente, o Apêndice A apresenta quatro artigos com contribuições para a área de Processamento de Fala não relacionadas com a aplicação de síntese de fala em reconhecimento automático de fala. O Apêndice A.1 apresenta contribuições para a tarefa de segmentação de sentenças em fala espontânea e comprometida por Doença de Alzheimer e Comprometimento Cognitivo Leve. Por sua vez, a Apêndice A.2 apresenta esforços para identificação de insuficiência respiratória em enunciados falados por pacientes com COVID-19, disponibilizando publicamente um *dataset* para esse fim. O Apêndice A.3 explora o uso de redes neurais pré-treinadas com áudio em larga escala para a identificação de COVID-19 em amostras de fala e tosse, utilizando um *dataset* multilíngue. Por fim, o Apêndice A.4 apresenta um novo método para treinamento de sistemas de verificação de locutores, denominado *Speech2Phone*, e compara com o estado da arte. O *Speech2Phone* alcançou resultados próximos ao estado da arte, utilizando 500 vezes menos dados durante o treinamento.

CAPÍTULO

2

FUNDAMENTAÇÃO TEÓRICA

Parte deste capítulo foi baseada no Trabalho de Conclusão de Curso (TCC) do autor (CASANOVA, 2019), sendo esse incrementado e atualizado com conceitos e trabalhos das áreas de Inteligência Artificial, Aprendizado de Máquina e Processamento de Fala.

Esse capítulo está organizado da seguinte forma. As Seções 2.1 a 2.6 apresentam conceitos gerais da área de Inteligência Artificial. Por sua vez, as Seções 2.7 a 2.10 apresentam conceitos da área de Inteligência Artificial largamente utilizados em Processamento de Fala. Por fim, as Seções 2.11 a 2.15 apresentam conceitos específicos de Processamento de Fala.

2.1 Redes Neurais Artificiais

De acordo com Russell e Norvig (2016), a primeira *Artificial Neural Network* (ANN) foi criada por Rosenblatt (1958) e foi baseada no algoritmo Perceptron. A partir disso, surgiram especulações na comunidade científica de que o modelo poderia vir a ser o pilar para estruturar um sistema de inteligência artificial forte. Entretanto, a área entrou em crise em 1960 devido ao trabalho proposto por Minsky e Selfridge (1960) com previsões pessimistas, devido à incapacidade do Perceptron resolver o problema do ou-exclusivo da lógica proposicional. Em 1989, Ramamoorthy e Shekhar (1989) propuseram um algoritmo de aprendizado adequado, denominado *Backpropagation*, ressurgindo assim o interesse na área. Desde então, o uso de ANNs vem crescendo muito, especialmente com o surgimento do *deep learning* (GOODFELLOW *et al.*, 2016). O interesse na área de Redes Neurais Artificiais oscilou muito com o passar do tempo, mas é inegável que ganhou espaço recentemente como um algoritmo *State-Of-The-Art* (SOTA) para diversas aplicações de aprendizado de máquina (GOODFELLOW *et al.*, 2016).

As ANNs foram projetadas com inspiração no funcionamento do cérebro humano. Portanto, o neurônio artificial foi inspirado no funcionamento do neurônio biológico. Do mesmo modo que o cérebro humano adquire conhecimento através de experiências, as ANNs também tendem a melhorar seu desempenho quando estão sendo treinadas para uma dada tarefa (HAYKIN *et al.*, 2009).

2.1.1 Neurônio Artificial

Uma ANN *feed-forward* é constituída por um número de unidades, conectadas por ligações. Estas unidades representam neurônios artificiais. Cada uma das ligações que conectam as unidades possuem um peso numérico associado a ela. As unidades são organizadas em camadas: a camada de entrada; as camadas intermediárias; e a camada de saída. As conexões entre neurônios de diferentes camadas simulam uma sinapse de um neurônio biológico. Durante o treinamento, os pesos são atualizados de forma a associar entradas às suas respectivas saídas (RUSSELL; NORVIG, 2016). Cada neurônio artificial possui três elementos, sendo eles um conjunto de sinapses, um agente somador e uma função de ativação, conforme mostra a Figura 1.

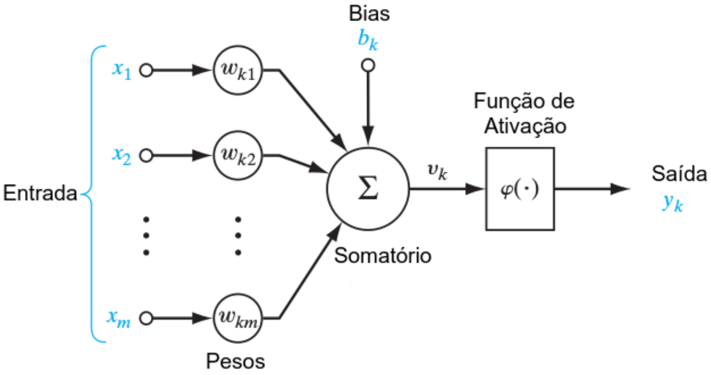


Figura 1 – Representação de um neurônio artificial

Fonte: Adaptada de Haykin *et al.* (2009).

No neurônio artificial as sinapses são representadas pelas n entradas de x_i, sendo que cada uma dessas entradas são ponderadas por um peso w_{ki}. O agente somador é responsável por acumular a atualização das sinapses $(x_i \cdot w_{ki})$ em u_k. Por fim, a função de ativação (φ) é responsável pela propagação da amplitude do sinal. O liminar de ativação (*bias*) tem o efeito de aumentar ou diminuir a entrada da função de ativação, e é denotado por b_k (HAYKIN *et al.*, 2009; RUSSELL; NORVIG, 2016).

O neurônio k, apresentado na Figura 1, pode ser representado matematicamente utilizando as equações 2.1 e 2.2. A Equação 2.1 apresenta o agente somador que acumula no vetor u_k as multiplicações entre as sinapses x_i e seus respectivos pesos w_i. E, por fim, na Equação 2.2 as operações são armazenadas em y_k que representa a saída desse neurônio.

$$u_k = \sum_{i=1}^{n} w_i x_i \tag{2.1}$$

$$y_k = \varphi(u_k + b_k) \tag{2.2}$$

2.1.2 Redes neurais artificiais Perceptron multicamadas

Os neurônios artificiais, apresentados na Seção 2.1.1, podem ser organizados em camadas, e essas camadas podem conter um ou mais neurônios. Essa organização de múltiplos neurônios em várias camadas pode ser denominada como rede neural artificial Perceptron multicamadas, também conhecida como *Multilayer Perceptron* (MLP). No tipo mais simples de rede MLP, a informação flui apenas da camada de entrada para a camada de saída; essas redes são conhecidas como redes *feed-forward*. Nesse tipo de rede não há informações da camada de saída sendo retro-alimentada para a entrada (HAYKIN, 2010).

Nas redes completamente conectadas *feed-forward* os neurônios de uma mesma camada não possuem nenhum tipo de ligação entre si. Entretanto, os neurônios de duas camadas vizinhas estão conectados entre si por sinapses (pesos), conforme mostra a Figura 2.

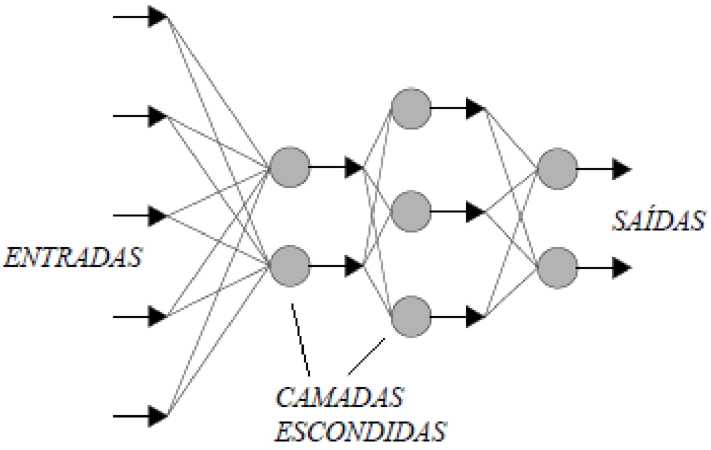


Figura 2 – Rede neural Perceptron multicamada *feed-forward* completamente conectada

Fonte: Adaptada de Rauber (2005).

Nas redes *feed-forward* a entrada é conectada a todos os neurônios da primeira camada intermediária e cada neurônio da última camada intermediária é conectado à camada de saída. Além disso, as camadas intermediárias são totalmente conectadas com suas camadas vizinhas (RUSSELL; NORVIG, 2016).

Segundo Hornik (1991), uma rede neural Perceptron multicamadas *feed-forward* que utiliza funções não lineares como ativação é capaz de aproximar qualquer função, desde que sejam usadas quantidades de neurônios necessárias. Por esse motivo, as redes multicamadas *feed-forward* são uma classe de aproximadores universais de funções. Entretanto, o uso de apenas ativações lineares limita o poder das redes multicamadas *feed-forward* para a aproximação de apenas funções linearmente separáveis. Por esta razão, é desejável que a ativação seja alterada para uma função não linear (HAYKIN, 2010).

2.2 Funções de ativação não lineares

Como discutido anteriormente, as funções de ativação não lineares desempenham um importante papel nas redes neurais artificiais. As funções de ativação definem a propagação das saídas dos neurônios nas redes neurais. Em geral, toda função de ativação recebe um único valor e executa uma operação fixa sobre ele, resultando assim em um outro valor. Existem muitas funções de ativação não lineares; nesta seção são apresentadas apenas algumas delas.

Sigmoide Logística. Segundo Haykin (2010), a função Sigmoide Logística é contínua e, portanto, diferenciável em todos os seus pontos. Essa função pode ser representada matematicamente pela Equação 2.3 e graficamente como mostra a Figura 3. A entrada da função é um número real x, e após a aplicação da função esse valor é mapeado para um intervalo entre 0 e 1 (GOODFELLOW *et al.*, 2016).

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{2.3}$$

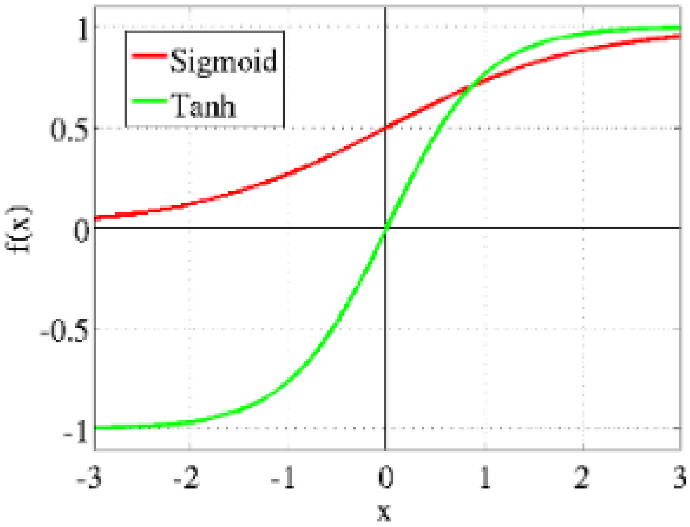


Figura 3 – Funções de ativação Sigmoide e Tanh

Fonte: Glorot, Bordes e Bengio (2011).

Apesar da Sigmoide Logística ser uma função amplamente utilizada em modelos neurais, ela possui duas desvantagens durante o treinamento de um modelo neural. A primeira é que quando a função se aproxima de zero ou de um, a derivada dessa função é próxima de zero, afetando os cálculos dos gradientes requeridos durante o treinamento da rede neural; isso é conhecido como problema da saturação (NIELSEN, 2015; TAN; LIM, 2019). A segunda é que a Sigmoide Logística nunca assume o valor zero como saída. As saídas com valor zero, em uma rede multicamadas *feed-forward*, permitem simplificar os cálculos na próxima camada, desse modo acelerando o treinamento da rede neural (LECUN; KANTER; SOLLA, 1991). Apesar dessas desvantagens, essa função ainda é muito utilizada, especialmente na camada de saída.

Tangente Hiperbólica. A função Tangente Hiperbólica é similar à Sigmoide Logística, porém, a escala de saída dela varia de -1 a 1, em vez de 0 a 1. Do mesmo modo que a Sigmoide Logística, ela também possui o problema da saturação (NIELSEN, 2015). Entretanto, a função Tangente Hiperbólica pode assumir o valor zero, podendo ser definida matematicamente pela Equação 2.4, onde σ representa a função Sigmoide Logística, e graficamente como mostra a Figura 3.

$$tanh(x) = 2\sigma(x) - 1 \tag{2.4}$$

ReLU. A função de ativação *Rectifier Linear Unit* (ReLU) (JARRETT *et al.*, 2009) é muito utilizada, especialmente em camadas convolucionais, que são apresentadas na Seção 2.3. Segundo Goodfellow *et al.* (2016), em redes neurais modernas recomenda-se como padrão o uso da ReLU como função de ativação. Essa função pode ser definida matematicamente pela Equação 2.5 e sua representação gráfica é mostrada na Figura 4 (ANTHIMOPOULOS *et al.*, 2016).

$$\phi(x) = max(0, x) \tag{2.5}$$

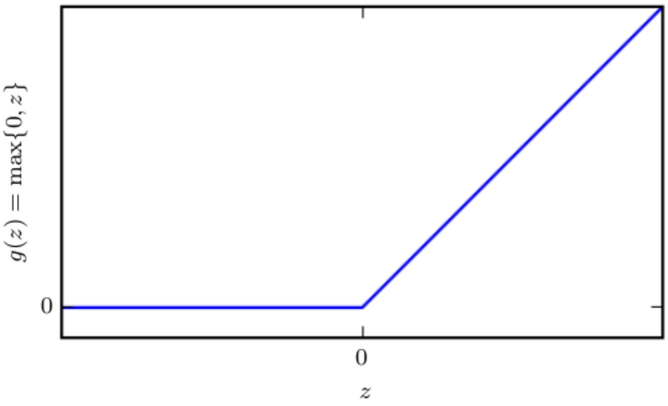


Figura 4 – Função de ativação *Rectifier Linear Unit*

Fonte: Goodfellow *et al.* (2016).

Uma desvantagem da função ReLU ficou conhecida como “Dying ReLU”. Esse problema ocorre quando um valor negativo é passado para a função e, consequentemente, a saída da função ReLU para esse valor será zero. Além disso, o gradiente também será zero, e deste modo o neurônio gera saída zero indefinidamente, interrompendo o aprendizado (PONTI *et al.*, 2021). Para melhorar o desempenho e solucionar este problema da função ReLU, foram propostas outras funções de ativação como a função LeakyReLU (MAAS; HANNUN; NG, 2013), CReLU (SHANG *et al.*, 2016), SELU (KLAMBAUER *et al.*, 2017), Swish (RAMACHANDRAN; ZOPH; LE, 2017) e Mish (MISRA, 2019).

2.3 Redes Neurais Convolucionais

As Redes Neurais Convolucionais, do inglês *Convolutional Neural Networks* (CNNs) (LECUN *et al.*, 1989), alcançaram sucesso em diferentes tarefas na área de visão computacional (GOODFELLOW *et al.*, 2016). Por exemplo, no reconhecimento de objetos (HE *et al.*, 2016; KRIZHEVSKY; SUTSKEVER; HINTON, 2017) e na identificação de faces (FARFADE; SABERIAN; LI, 2015; DANG *et al.*, 2018; MO; CHEN; LUO, 2018). Além disso, também obtiveram êxito em tarefas como reconhecimento automático de fala (AMODEI *et al.*, 2016; LI *et al.*, 2019a; KRIMAN *et al.*, 2020), verificação de locutores (CHUNG *et al.*, 2020a) e síntese de fala (TACHIBANA; UENOYAMA; AIHARA, 2018; PING *et al.*, 2018; KIM *et al.*, 2020). As CNNs provaram ser muito eficientes como extratores de características (RAZAVIAN *et al.*, 2014).

Segundo Haykin (2010), uma CNN é uma rede Perceptron multicamadas projetada especificamente para reconhecer formas bidimensionais com um alto grau de invariância à translação, dimensionamento, enviesamento e outras formas de distorção.

Além disso, ao contrário das redes neurais artificiais Perceptron multicamadas, as CNNs preservam as relações locais em uma dada vizinhança, e aprendem uma representação interna da entrada usando filtros (NIELSEN, 2015).

Uma CNN típica possui três componentes principais: camada convolucional (*convolutional layer*), descrita na Seção 2.3.1, função não linear (*nonlinear function*) como a ReLU, descrita previamente na Seção 2.2, e camada de *pooling* (*pooling layer*), descrita na Seção 2.3.2 (NIELSEN, 2015; GOODFELLOW *et al.*, 2016).

2.3.1 Camada Convolucional

A camada convolucional é o principal elemento das CNNs. Ela implementa o compartilhamento de pesos que possibilita trabalhar com grandes entradas. As camadas convolucionais, diferentemente das camadas completamente conectadas, conseguem levar em conta a estrutura da entrada, podendo assim trabalhar em vizinhanças bem definidas (NIELSEN, 2015).

Segundo Goodfellow *et al.* (2016), a camada convolucional realiza várias convoluções em paralelo para produzir um conjunto de ativações lineares.

A camada convolucional consiste em vários neurônios responsáveis por extrair diferentes recursos das sub-regiões da entrada (HIJAZI; KUMAR; ROWEN, 2015). Nas CNNs, as entradas podem ser de diferentes formas. Existem as camadas convolucionais 1D, que podem ser usadas, por exemplo, para processar áudio em formato wav; nesse caso, o eixo de convolução corresponde ao tempo. Por outro lado, camadas 2D podem ser usadas para processar espectrogramas, discutidos na Seção 2.11. Em um espectrograma, as linhas correspondem a diferentes frequências e as colunas correspondem a diferentes pontos no tempo. Nesse caso, é possível aplicar convolução tanto no eixo do tempo, para tornar o modelo equivariante às mudanças no tempo, quanto no eixo das frequências tornando o modelo equivariante à frequência. Por fim, camadas 3D podem ser utilizadas em dados volumétricos, como tomografias computadorizadas (GOODFELLOW *et al.*, 2016).

Em uma CNN com camadas 2D, neurônios são usados para extrair uma determinada característica e são agrupados em um filtro de estrutura bidimensional. Além disso, cada camada convolucional consiste em vários filtros sobrepostos, gerando uma estrutura tridimensional.

Em uma camada convolucional é necessário especificar a quantidade de filtros, seus tamanhos e o *stride*, que define o tamanho da vizinhança que os neurônios de cada camada processará (VARGAS; PAES; VASCONCELOS, 2016). A camada convolucional é assim chamada devido à operação de convolução realizada pela mesma. O processo de convolução nada mais é do que deslizar (convolver) o filtro através da entrada, realizando em cada uma das posições um produto interno através do qual é obtido um escalar linear (NIELSEN, 2015). Após a aplicação do processo de convolução, obtém-se um mapa de ativações lineares. As dimensões desse mapa dependem do tamanho da entrada, tamanho do *stride*, tamanho do *kernel* e do *zero-padding*. Em uma camada convolucional, se o *stride* tem valor 1, os filtros são deslocados cada vez de coluna em coluna ou de linha em linha, ou seja, caminham um passo de cada vez. Ao se empilhar camadas convolucionais, o tamanho da representação diminui sucessivamente em cada camada até desaparecer. O *zero-padding* pode ser utilizado para adicionar bordas de zeros na entrada da camada para manter a saída com o tamanho igual ao da entrada após a aplicação da convolução (NIELSEN, 2015). A Figura 5 apresenta uma camada convolucional em que o tamanho da entrada é 28×28, o tamanho do *kernel* é 5×5, e o *stride* é 1, resultando em um mapa de ativação de tamanho 24×24.

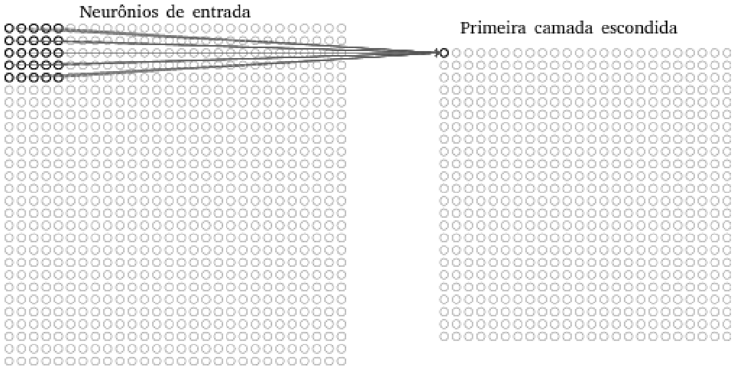


Figura 5 – Exemplo de uma camada convolucional

Fonte: Adaptada de Nielsen (2015).

2.3.2 Camada de pooling

Como discutido na Seção 2.3.1, as camadas convolucionais naturalmente já diminuem a dimensionalidade de suas entradas, tratando, desse modo, do problema da maldição da dimensionalidade (HAYKIN *et al.*, 2009), comumente presente em dados de grande dimensionalidade como imagens e sinais de áudio. Contudo, mais reduções podem ser necessárias (MAZZA, 2017).

A camada de *pooling* geralmente é adicionada logo após a função de ativação de uma camada convolucional, reduzindo a dimensionalidade da entrada das camadas seguintes. Existem vários tipos de *pooling*, sendo que um dos mais populares é conhecido como *max-pooling* (NIELSEN, 2015). Uma camada de *max-pooling*, como o nome já indica, retorna os valores máximos obtidos em seu filtro. A camada de *max-pooling* não realiza nenhum aprendizado, em vez disso, ela reduz o número de parâmetros a serem aprendidos nas camadas seguintes (PENHA; CASTRO, 2017). Por exemplo, uma camada de *max-pooling* com um *kernel* de tamanho 2 × 2 e um *stride* de 2 pode reduzir um mapa de ativação de 24 × 24 para 12 × 12, conforme ilustrado na Figura 6.

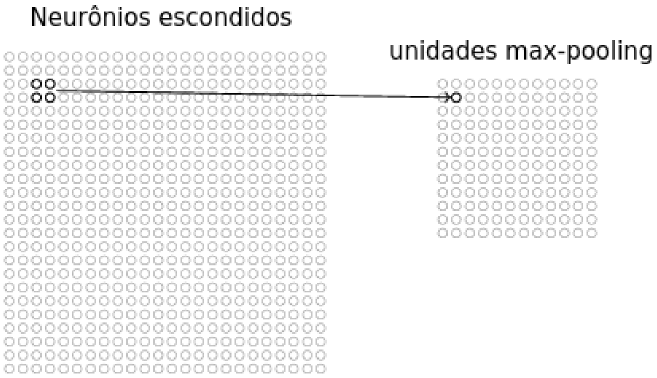


Figura 6 – Exemplo de uma camada *max-pooling*

Fonte: Adaptada de Nielsen (2015).

2.4 Redes Neurais Recorrentes

As *Recurrent Neural Networks* (RNNs) (MEDSKER; JAIN, 2001), ou Redes Neurais Recorrentes em português, foram criadas por Rumelhart, Hinton e Williams (1985) com o propósito de aprender sequências de caracteres. As RNNs ganharam popularidade com os avanços do *deep learning* e com o aumento do poder computacional das unidades de processamento gráfico. Desde então, as RNNs têm sido aplicadas em diferentes tarefas como na tradução automática (FIRAT; CHO; BENGIO, 2016), modelagem de língua (MIKOLOV *et al.*, 2010; MIKOLOV *et al.*, 2013), síntese de fala (WANG *et al.*, 2017; SHEN *et al.*, 2018), identificação de locutor (REN *et al.*, 2016) e reconhecimento automático de fala (GRAVES; JAITLY; MOHAMED, 2013), entre outras.

As RNNs, diferentemente das redes *Multilayer Perceptron*, além de operarem com base nas entradas, também possuem estados internos. Os estados internos têm como função codificar as informações anteriores na sequência temporal já processada pela RNN. Por exemplo, uma entrada fornecida a uma RNN em um instante de tempo t pode alterar o comportamento dessa mesma rede em um momento $t+k$, considerando $k > 0$. Dessa forma, em uma RNN a saída da rede não depende apenas da entrada atual, mas da entrada atual e do estado atual. Essa característica permite realizar computação dependente de contexto e aprender dependências de longo prazo (YU; DENG, 2016).

Podemos dividir uma RNN em duas partes principais. A primeira parte é uma função que é responsável por mapear a entrada e o estado anterior para o estado atual. A segunda parte é uma função que mapeia o estado atual para a saída da rede. Para simplificar a representação das funções será considerado que a rede opera em uma sequência que contém vetores $\mathbf{x}^{(t)} \in \mathbb{R}^D$ com o índice de intervalo de tempo t variando de 1 a T. Deste modo, podemos definir a primeira função com a Equação 2.6 e a segunda com a Equação 2.7 (YU; DENG, 2016; GOODFELLOW *et al.*, 2016).

$$h^{(t)} = tanh(\mathbf{W}_{xh}^T \mathbf{x}^{(t)} + \mathbf{W}_{hh}^T \mathbf{h}^{(t-1)} + b_h) \tag{2.6}$$

$$y^{(t)} = \mathbf{W}_{hy}^T \mathbf{h}^{(t)} + b_y \tag{2.7}$$

Considere nas equações $\mathbf{y}^{(t)}$ como a saída da rede no intervalo de tempo t e que o $\mathbf{h}^{(t)}$ representa o estado interno da rede no instante t. O tensor W_{xh}^T representa os pesos que conectam o vetor de entrada $x^{(t)}$ ao vetor do estado interno da rede $h^{(t)}$. O tensor W_{hh}^T conecta o vetor $h^{(t-1)}$ que é o estado interno produzido pela entrada anterior $x^{(t-1)}$ ao estado interno atual $h^{(t)}$. O vetor b_h representa os *biases* da camada intermediária e *tanh* representa a função tangente hiperbólica. Por fim, o tensor W_{hy}^T conecta o vetor $h^{(t)}$ ao vetor de saída $y^{(t)}$ juntamente com os *biases* (b_y) (YU; DENG, 2016). Para facilitar o entendimento, a Figura 7 apresenta uma representação gráfica do processo, onde a linha tracejada representa o retorno do estado interno ($h^{(t-1)}$) produzido pela entrada anterior ($x^{(t-1)}$) ao estado interno atual ($h^{(t)}$).

Existem outros tipos de RNNs, como *Long Short-Term Memory* (LSTM) (HOCHREITER; SCHMIDHUBER, 1997) e *Gated Recurrent Unit* (GRU) (CHUNG *et al.*, 2014), porém não serão abordadas nesse trabalho. Para mais detalhes, verifique os livros de Goodfellow *et al.* (2016) e Zhang *et al.* (2020).

2.5 Deep learning

Os modelos baseados em *deep learning*, ou aprendizado profundo, são atualmente o estado da arte em grande parte dos problemas que são possíveis de serem resolvidos com o aprendizado de máquina. Por exemplo, diversos problemas de visão computacional como reconhecimento de objetos (TOUVRON *et al.*, 2020; SRINIVAS *et al.*, 2021) e identificação

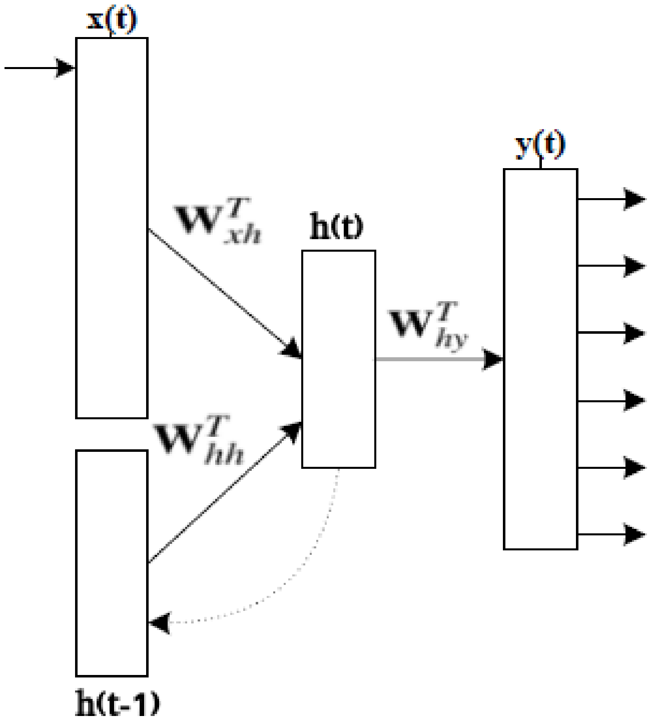


Figura 7 – Topologia de uma RNN com uma camada recorrente

Fonte: Adaptada de Guo (2013).

de faces (HUANG *et al.*, 2020; SHI *et al.*, 2020; YANG *et al.*, 2021). Além disso, na área da fala, temos o reconhecimento automático de fala (BAEVSKI *et al.*, 2020; KRIMAN *et al.*, 2020; HSU *et al.*, 2021), verificação de locutores (CHUNG *et al.*, 2020b; ZHOU; ZHAO; WU, 2021) e síntese de fala (VALLE *et al.*, 2020; KIM *et al.*, 2020; KIM; KONG; SON, 2021).

O grande sucesso do *deep learning* deve-se principalmente a dois motivos. Primeiro, a disponibilidade de *datasets* com grande quantidade de dados, por exemplo, na visão computacional *datasets* com milhões de imagens (DENG *et al.*, 2009; RUSSAKOVSKY *et al.*, 2015) e na área da fala *datasets* com milhares de horas (PANAYOTOV *et al.*, 2015; CHUNG; NAGRANI; ZISSERMAN, 2018; ZEN *et al.*, 2019). Segundo, a grande evolução computacional, que reduziu o tempo necessário para realizar o treinamento de modelos com esses grandes *datasets*.

Deep learning atraiu grande atenção de pesquisadores devido ao bom desempenho de modelos neurais no *dataset* Imagenet (DENG *et al.*, 2009). Krizhevsky, Sutskever e Hinton (2012) propuseram o modelo AlexNet, que foi um dos primeiros modelos baseados no *deep learning* a obter um desempenho promissor no *dataset* Imagenet, desse modo atraindo grande atenção para a área.

As *Deep Neural Networks* (DNNs) basicamente são redes *Multilayer Perceptron* com a adição de mais camadas intermediárias, dependendo das necessidades do problema a ser solucionado (YU; DENG, 2016). Uma das primeiras DNNs foi proposta por Hinton, Osindero e Teh (2006), e tem apenas três camadas intermediárias. Do mesmo modo que as DNNs são redes neurais com várias camadas intermediárias, as *Deep Convolutional Neural Networks* (DCNNs) (MALLAT, 2016) e as *Deep Recurrent Neural Networks* (DRNN) (PASCANU *et al.*, 2014) são baseadas respectivamente nas CNNs e RNNs. Quanto maior a profundidade e o número de neurônios que uma rede possui, maior á quantidade de dados necessária para o aprendizado. Além disso, experimentos empíricos têm mostrado que, quanto maior e mais adequado o *dataset*, maior também será a possibilidade da rede generalizar novas instâncias nunca vistas no treinamento (GOODFELLOW *et al.*, 2016). No entanto isso traz desafios para o treinamento, sendo necessário recorrer à estratégias e arquiteturas para permitir a convergência dos modelos (PONTI *et al.*, 2021).

2.6 Redes neurais convolucionais com skip connections

Como discutido na Seção 2.3, a eficiência das CNNs é inegável. Entretanto, com o advento do *deep learning* e o aumento da profundidade das redes, surgiu um novo problema denominado *vanishing gradient* (NIELSEN, 2015). O *vanishing gradient* ocorre quando a informação da entrada ou o próprio gradiente é propagado por várias camadas consecutivas e a informação começa a “desaparecer”, assumindo um valor cada vez menor até chegar ao final (entrada) ou ao início (gradiente) da rede neural artificial (HUANG *et al.*, 2017). As DCNNs geralmente são divididas em blocos, cada um composto por um número x de camadas convolucionais, podendo incluir também camadas de normalização como *layer normalization* (BA; KIROS; HINTON, 2016) ou *batch normalization* (BJORCK *et al.*, 2018).

Para tentar minimizar os efeitos do *vanishing gradient*, foram propostas alterações nas topologias dos modelos convolucionais adicionando conexões que ligam a entrada de um bloco à entrada de outro bloco. Essas conexões ficaram conhecidas como *skip connections*, ou ainda como conexões residuais. Com o uso de *skip connections* na topologia da rede, a informação desaparece de uma forma mais lenta. Isso ocorre devido a informação também ser propagada diretamente da entrada do bloco anterior, de certa forma pulando a perda de informação que ocorreria.

Srivastava, Greff e Schmidhuber (2015) propuseram as *highway networks*, que foram uma das primeiras abordagens a mostrar a viabilidade de treinar redes neurais de ponta a ponta com mais de 100 camadas (HUANG *et al.*, 2017). As *highway networks* foram inspiradas nas redes recorrentes LSTM. Na saída de cada bloco das *highway networks* existe uma porta de transformação (*transform gate*). Além disso, a entrada de cada bloco é conectada diretamente à porta de transformação desse mesmo bloco por meio de uma *skip connection*. Durante o treinamento do modelo, a porta de transformação decide quanto de informação da saída do bloco irá passar para o próximo bloco e também quanto de informação da entrada desse bloco, que chegou por meio da *skip connection*, será propagada para o próximo bloco. Por exemplo, se a porta de transformação predizer o valor de $0,7$, significa que 70% da informação da saída do bloco será propagada para o próximo bloco e consequentemente os 30% restantes serão propagados da *skip connection*. Além disso, as informações da saída do bloco e da *skip connection* são misturadas utilizando a operação de soma. Essa configuração é interessante pois permite que o bloco atue tanto como um bloco convencional quanto como um bloco bastante simples que propaga suas entradas para a próxima camada sem modificá-las, ou ainda como um meio termo entre os dois tipos de blocos, propagando para o próximo bloco informações da entrada do bloco e da saída do bloco misturadas. A Figura 8 representa graficamente dois blocos de uma *highway networks*, em que $*$ representa a porta de transformação.

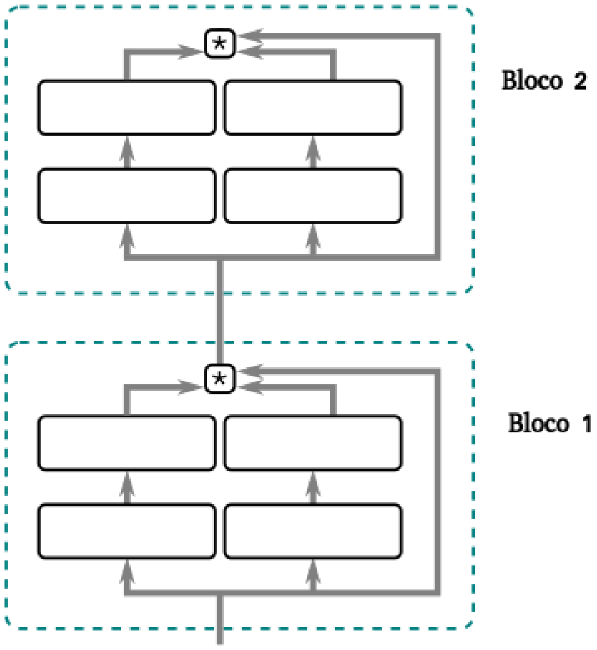


Figura 8 – Blocos de uma *highway network*

Fonte: Adaptada de Greff, Srivastava e Schmidhuber (2016).

He *et al.* (2016) propuseram a ResNet. Nessa abordagem, cada bloco é ligado com o seu antecessor através de uma *skip connection*. Diferentemente das *highway networks* que possuem uma porta de transformação que controla a quantidade de informação da *skip connection* que é propagada para o próximo bloco, na ResNet toda a informação da saída do bloco e toda a informação da *skip connection* é propagada para o próximo bloco. Além disso, como nas *highway networks* as informações da saída do bloco e da *skip connection* são misturadas utilizando a operação de soma. A Figura 9 apresenta a representação gráfica de dois blocos da ResNet, sendo que $+$ denota a operação de soma.

Por outro lado, Huang *et al.* (2017) propuseram a DenseNet. Diferentemente da ResNet, todos os blocos da DenseNet estão diretamente conectados entre si, ou seja, a entrada de um bloco recebe uma entrada adicional de todos os blocos anteriores, como pode ser visto na Figura 10. Além disso, na DenseNet as *skip connections* são combinadas com a saída do bloco antecessor por meio de uma concatenação, em vez da soma como realizado na *highway* e ResNet. O uso da

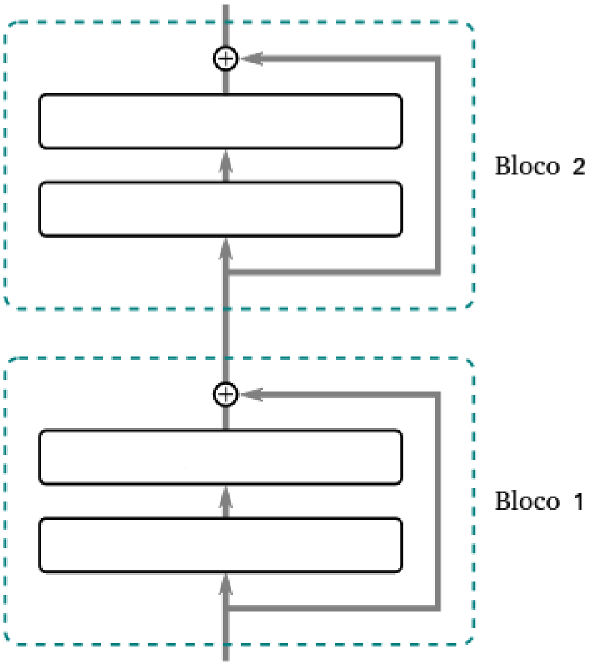


Figura 9 – Blocos de uma ResNet

Fonte: Adaptada de Greff, Srivastava e Schmidhuber (2016).

concatenação em vez da soma tem suas vantagens entretanto, aumenta de forma significativa o uso de memória do modelo.

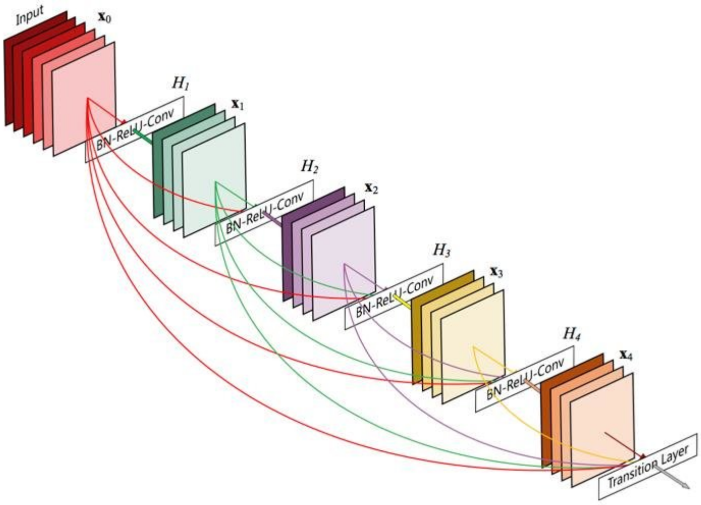


Figura 10 – Arquitetura do modelo DenseNet

Fonte: Huang *et al.* (2017).

2.7 Mecanismo de Atenção

A capacidade de prestar atenção em apenas uma fração das informações tem significado evolutivo, permitindo que os seres humanos vivam e tenham sucesso (ZHANG *et al.*, 2020). Nessa seção, será apresentado o problema que os mecanismos de atenção buscam resolver nas redes neurais artificiais, bem como o funcionamento do mecanismo de atenção em uma rede neural.

Sutskever, Vinyals e Le (2014) propuseram um modelo que mapeia uma sequência para outra sequência; essa abordagem ficou conhecida como *seq2seq*. Essa abordagem foi inicialmente pensada para ser usada na modelagem para tratar línguas naturais. Em geral, o objetivo dos modelos *seq2seq* é transformar uma sequência de entrada em uma nova sequência, em que ambas as sequências podem ter comprimentos diferentes. Os modelos *seq2seq* geralmente possuem uma arquitetura codificador-decodificador (*encoder-decoder*), e os trabalhos de Colombo *et al.* (2020), Shen *et al.* (2018), Cho *et al.* (2014) são exemplos dessa abordagem. Nessa arquitetura, o codificador transforma a sequência de entrada compactando-a em um vetor de contexto de tamanho fixo, também conhecido como *sentence embedding*. Por sua vez, o decodificador recebe o vetor de contexto e o transforma na saída (WENG, 2018a).

Um problema com a abordagem codificador-decodificador é que o codificador do modelo precisa ser capaz de comprimir todas as informações necessárias da entrada em um vetor de contexto de tamanho fixo. Essa compressão se torna difícil quando se tem sequências muito longas. Da mesma forma que para os primatas a atenção permite focalizar e se concentrar, direcionando a atenção para objetos de interesse, o fato de uma rede neural conseguir prestar a atenção pode auxiliar na solução de problemas, pois a atenção pode permitir focar nas partes importantes da sequência de entrada enquanto estiver predizendo determinada parte da sequência. Por exemplo, no caso da tradução automática o modelo pode prever uma palavra baseando-se apenas nas palavras da origem mais relevantes (ZHANG *et al.*, 2020).

Buscando uma solução para o problema da abordagem codificador-decodificador e inspirando-se em como o mecanismo de atenção humano funciona, Graves (2013) projetou um mecanismo de atenção diferenciável para a geração de caligrafia utilizando uma sequência de texto, mostrando que era possível alinhar caracteres de texto com traços de canetas muito mais longos.

Os humanos prestam atenção muitas vezes de forma involuntária como, por exemplo, ao olhar para uma mesa com dois livros e uma xícara vermelha, a cor vermelha da xícara chamará atenção de forma involuntária, pois mesmo não tendo nenhum comando para os olhos prestarem atenção na xícara isso ocorre. No mecanismo de atenção, o mecanismo que permite prestar atenção de forma involuntária é chamado de *keys*. Os humanos também são capazes de prestar atenção em determinados objetos de forma voluntária, em que é escolhido em qual objeto prestar atenção. Por exemplo, se na mesa deseja-se prestar atenção em um dos dois livros, a atenção que estava direcionada na xícara é movida para o livro. Essa informação de onde prestar a atenção no mecanismo de atenção é chamada de *query*. E, por fim, os objetos/informações sensoriais de entrada são chamados de *values*. De modo geral, os *values* (entrada sensorial) são emparelhados com as *keys*, que podem ser pensadas como a sugestão involuntária dessa entrada sensorial (ZHANG *et al.*, 2020). A Figura 11 apresenta uma representação gráfica do mecanismo de atenção.

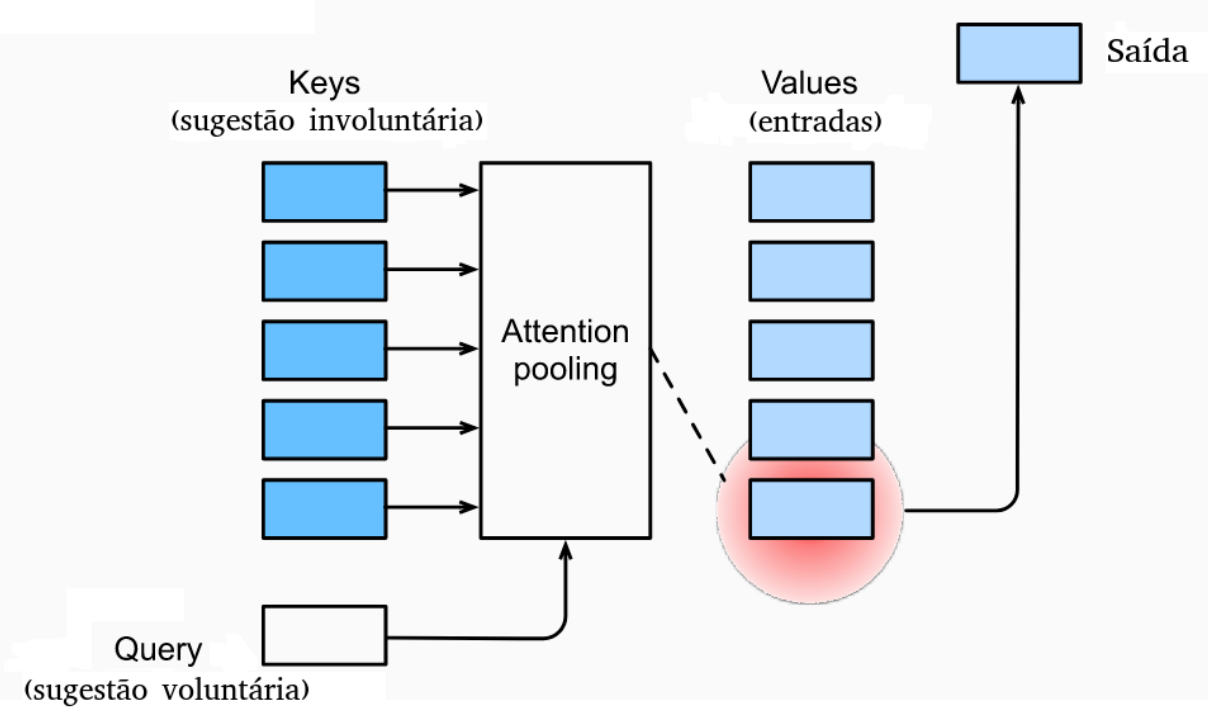


Figura 11 – Representação gráfica de um mecanismo de atenção

Fonte: Adaptada de Zhang *et al.* (2020).

Em geral, o *attention pooling* varia conforme a função de *score* nos diferentes tipos de atenção. Para facilitar, daqui em diante assume-se uma representação genérica f_{score} independente da função de *score* utilizada e as principais funções de *score* serão apresentadas na Seção 2.7.1.

A saída de um mecanismo de atenção, também conhecida como vetor de contexto, pode ser definida pela Equação 2.8, considerando f_{score} a função de *score* escolhida, uma *query* $\mathbf{q} \in \mathbb{R}^q$, uma *key* $\mathbf{k} \in \mathbb{R}^k$, um *value* $\mathbf{v} \in \mathbb{R}^v$ e por fim $softmax$ representando a função de ativação Softmax (ZHANG *et al.*, 2020). As *keys*, *queries* e *values* podem assumir diferentes valores dependendo do problema que está sendo explorado.

$$\text{Attention}(\mathbf{q},\mathbf{k},\mathbf{v}) = \mathit{softmax}(f_{score}(\mathbf{q},\mathbf{k}))v \tag{2.8}$$

2.7.1 Funções de score

Como discutido anteriormente, o *attention pooling* varia conforme a função de *score* utilizada nos diferentes tipos de atenção. Nessa seção são listadas algumas das principais funções de *score*, considerando uma *query* $\mathbf{q} \in \mathbb{R}^q$ e uma *key* $\mathbf{k} \in \mathbb{R}^k$.

Additive Attention. No mecanismo de atenção proposto por Bahdanau, Cho e Bengio (2015), o *score* do alinhamento da atenção é parametrizado por uma rede *feed-forward* com uma única camada intermediária e essa rede é treinada em conjunto com outras partes do modelo. Esse tipo de atenção ficou conhecido como *Additive Attention* (VASWANI *et al.*, 2017). A função de *score* aditiva pode ser definida pela Equação 2.9, onde $\mathbf{W}_q \in \mathbb{R}^{h\times q}$, $\mathbf{W}_k \in \mathbb{R}^{h\times k}$ e $\mathbf{w}_v \in \mathbb{R}^h$ são parâmetros aprendidos durante o treinamento e h o número de neurônios na camada oculta (ZHANG *et al.*, 2020):

$$f_{score}(\mathbf{q},\mathbf{k}) = \mathbf{w}_v^\top \tanh(\mathbf{W}_q\mathbf{q} + \mathbf{W}_k\mathbf{k}) \tag{2.9}$$

General Attention. A *General Attention* foi proposta por Luong, Pham e Manning (2015). A função de *score* utilizada por esse mecanismo de atenção pode ser definida com a Equação 2.10, considerando $\mathbf{W}_k \in \mathbb{R}^{h\times k}$ parâmetros aprendidos durante o treinamento e h o número de neurônios na camada oculta:

$$f_{score}(\mathbf{q},\mathbf{k}) = \mathbf{q}^\top \mathbf{W}_k\mathbf{k} \tag{2.10}$$

Dot-Product Attention. A *Dot-Product Attention* também foi proposta por Luong, Pham e Manning (2015). Essa função de *score* é computacionalmente mais eficiente que as anteriormente citadas. Ela é basicamente o produto escalar entre as *Queries* e as *Keys*. Entretanto, a operação de produto escalar requer que as *Queries* e as *Keys* tenham o mesmo comprimento de vetor (ZHANG *et al.*, 2020). A função de *score* utilizada por esse mecanismo de atenção pode ser definida com a Equação 2.11:

$$f_{score}(\mathbf{q},\mathbf{k}) = \mathbf{q}^\top \mathbf{k} \tag{2.11}$$

Scaled Dot-Product Attention. Apesar da *Dot-Product Attention* ser muito eficiente e aplicável para problemas onde as *Queries* e as *Keys* tenham o mesmo tamanho de vetor, se o tamanho desse vetor for d e todos os elementos das *Queries* e as *Keys* forem variáveis aleatórias independentes com média zero e variância unitária, o produto escalar de ambos os vetores terá média zero e uma variância d. Buscando garantir que a variância do produto escalar permaneça um, independentemente do tamanho do vetor, Vaswani *et al.* (2017) propuseram a divisão do produto escalar pela raiz quadrada de d ($\sqrt{d}$). Portanto, a função de *score* utilizada por esse mecanismo de atenção pode ser definida com a Equação 2.12:

$$f_{score}(\mathbf{q},\mathbf{k}) = \mathbf{q}^\top \mathbf{k}/\sqrt{d} \tag{2.12}$$

2.7.2 Multi-head Attention

Vaswani *et al.* (2017), buscando combinar o conhecimento de diferentes comportamentos de um mesmo mecanismo de atenção, propuseram a *multi-head attention*. Segundo Vaswani *et al.* (2017), a *multi-head attention*, ao contrário da atenção com uma única cabeça (*head*), permite que o modelo atenda conjuntamente às informações de diferentes subespaços de representação em diferentes posições. Na *multi-head attention* em vez de realizar um único agrupamento de atenção, as *queries*, *keys* e *values* são transformadas com *h* projeções lineares aprendidas de forma independente (ZHANG *et al.*, 2020). As *queries*, *keys*, e *values* após serem transformadas pelas *h* projeções, são passadas para um conjunto de *attention pooling* e computadas em paralelo. Por fim, as saídas dos *attention pooling* são concatenadas e alimentadas para uma outra projeção linear produzindo a saída final do mecanismo. A Figura 12 apresenta uma representação gráfica de uma *multi-head attention* utilizando *attention pooling* com a função de *score Scaled Dot-Product*, onde *concat* indica concatenação, *linear* representa uma matriz de pesos treináveis com ativação linear (projeção linear), *h* representa o número de cabeças (*heads*) de atenção, *V* representa os *values*, *K* representa as *keys* e *Q* representa a *query*.

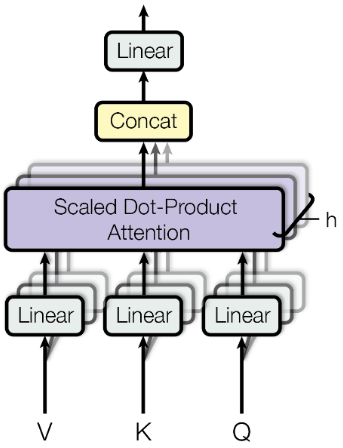


Figura 12 – Representação gráfica da *multi-head attention*

Fonte: Vaswani *et al.* (2017).

2.7.3 Self-Attention

Self-attention (VASWANI *et al.*, 2017; LIN *et al.*, 2017) consiste basicamente em um mecanismo de atenção como os vistos anteriormente onde as *keys*, *values* e *queries* possuem o mesmo valor. Nesse tipo de atenção as *queries* são passadas três vezes para o mecanismo de atenção, substituindo os *values* e *keys*. Portanto, cada *query* atende a todos os pares de *values*-*keys* e gera uma saída de atenção (ZHANG *et al.*, 2020). Como as *queries*, *keys* e *values* vêm do mesmo lugar, essa configuração realiza *self-attention*, também conhecida como *intra-attention* (PARIKH *et al.*, 2016).

2.8 Transformers

Os *Transformers* foram propostos por Vaswani *et al.* (2017), e desde então foram aplicados com sucesso em diferentes tarefas, por exemplo, em modelos de língua (DEVLIN *et al.*, 2019), aprendizado de representações visuais para tarefas de classificação, recuperação e síntese (DOSOVITSKIY *et al.*, 2020; LIU *et al.*, 2020; RIBEIRO *et al.*, 2020), e síntese de fala (LI *et al.*, 2019b; KIM *et al.*, 2020; KIM; KONG; SON, 2021). Os autores dos *Transformers* introduziram uma série de melhorias nos mecanismos de atenção, como a *multi-head attention* e a função de *score Scaled Dot-Product*, já apresentadas anteriormente. Os *Transformers* foram os primeiros modelos a tornarem possível realizar a modelagem *seq2seq* sem o uso de redes neurais recorrentes. Eles são inteiramente construídos sobre os mecanismos de *self-attention* sem usar nenhuma camada recorrente.

A arquitetura proposta por Vaswani *et al.* (2017) é uma arquitetura seq2seq que possui um codificador e um decodificador com blocos diferentes. Nessa monografia será abordada apenas a arquitetura do bloco do codificador, que foi utilizada, por exemplo, no modelo de língua BERT (DEVLIN *et al.*, 2019) e no modelo de síntese de fala GlowTTS (KIM *et al.*, 2020). No trabalho original, o codificador é composto por 6 blocos seguidos. Cada um desses blocos possui duas camadas seguidas de *layer normalization* (BA; KIROS; HINTON, 2016). A primeira camada é uma de *self-attention multi-head* com 8 cabeças que utiliza a função de *score Scaled Dot-Product*. A segunda é um sub-bloco *feed-forward* que é composto por duas camadas *feed-forward* seguidas, podendo ser duas camadas convolucionais 1D (KIM *et al.*, 2020) ou duas camadas densas (DEVLIN *et al.*, 2019). Similar à ResNet, apresentada anteriormente na Seção 2.6, os autores usam uma *skip connection* ligando a entrada e a saída tanto da camada de atenção quanto do sub-bloco *feed-forward*, em ambas os dados da saída das camadas são misturados com o uso da operação de soma e após é aplicado o *layer normalization*. A Figura 13 apresenta a representação de um bloco do codificador da arquitetura Transformer. Considere *Add* representando a operação de soma e *Norm* a *layer normalization*.

2.9 Autoencoders e Variational Autoencoders

Um *autoencoder* é uma rede neural artificial que é treinada para tentar copiar a sua entrada para a sua saída (GOODFELLOW *et al.*, 2016). Ele foi proposto por Hinton e Salakhutdinov (2006) para tentar reduzir a dimensionalidade de dados utilizando redes neurais e aplicado, por exemplo, para comprimir uma imagem em uma representação mais compacta.

O *autoencoder* tem duas partes principais: o codificador (*encoder*) e o decodificador (*decoder*). O codificador tem a função de transformar a entrada x, que pode ser por exemplo uma imagem, em uma representação comprimida z de baixa dimensionalidade da entrada, que é geralmente chamada de *bottleneck*. Por sua vez, o decodificador utilizando-se dessa representação comprimida (z), precisa aprender a reconstruir a entrada. O codificador e o decodificador são

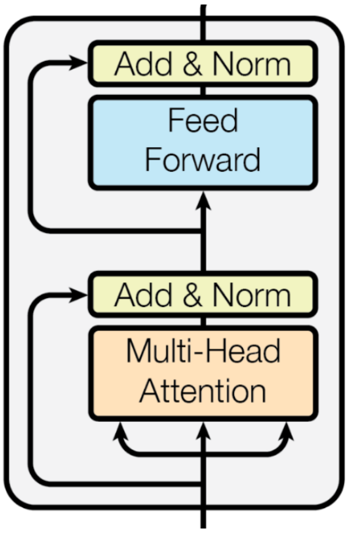


Figura 13 – Bloco Transformer

Fonte: Vaswani *et al.* (2017).

treinados em conjunto e podem ser constituídos de diferentes tipos de camadas, por exemplo, convolucionais e recorrentes, dependendo do problema a ser explorado. Em processamento de imagens, geralmente o codificador é composto por camadas convolucionais e o decodificador é composto por camadas de de-convolução (SHI *et al.*, 2016). Para guiar o aprendizado é possível utilizar diferentes funções, e essas funções devem permitir calcular a diferença entre a entrada e a saída predita pelo *autoencoder*. Exemplos de funções que são empregadas na tarefa são a L1 *loss* e a *Mean Squared Error* (MSE) (GOODFELLOW; BENGIO; COURVILLE, 2016). A Figura 14 apresenta a arquitetura geral de um *autoencoder*, considerando x como a entrada e x' como a saída predita pelo *autoencoder*.

Apesar da abordagem do *autoencoder* tradicional ser interessante e eficiente para a compressão de dados, essa abordagem tem uma limitação. Apesar do codificador transformar uma entrada em uma representação comprimida e o decodificador reconstruir a entrada, no *autoencoder* tradicional não é possível produzir qualquer conteúdo novo, por exemplo, gerar uma representação aleatória da representação comprimida e obter uma nova instância para utilizar em abordagens de aumento de dados. Isso porque esses modelos memorizam os dados de treinamento de forma que a reconstrução se assemelha aos dados vistos mesmo quando instâncias diferentes são apresentadas (CAVALLARI; RIBEIRO; PONTI, 2018). Pensando nesse problema e inspirando-se na Máquina de Helmholtz (DAYAN *et al.*, 1995) Kingma e Welling (2013) propuseram o *variational autoencoder* (VAE).

A ideia do *variational autoencoder* é diferente do *autoencoder* tradicional. Em vez de mapear a entrada em um vetor de tamanho fixo, ele mapeia a entrada em uma distribuição. Além disso, para ser possível a aplicação do *autoencoder* em geração de dados, apenas transformar a entrada em uma distribuição não é suficiente. É também necessário garantir que o espaço latente seja regular o suficiente. Por esse motivo, durante o treinamento de um VAE o espaço latente é

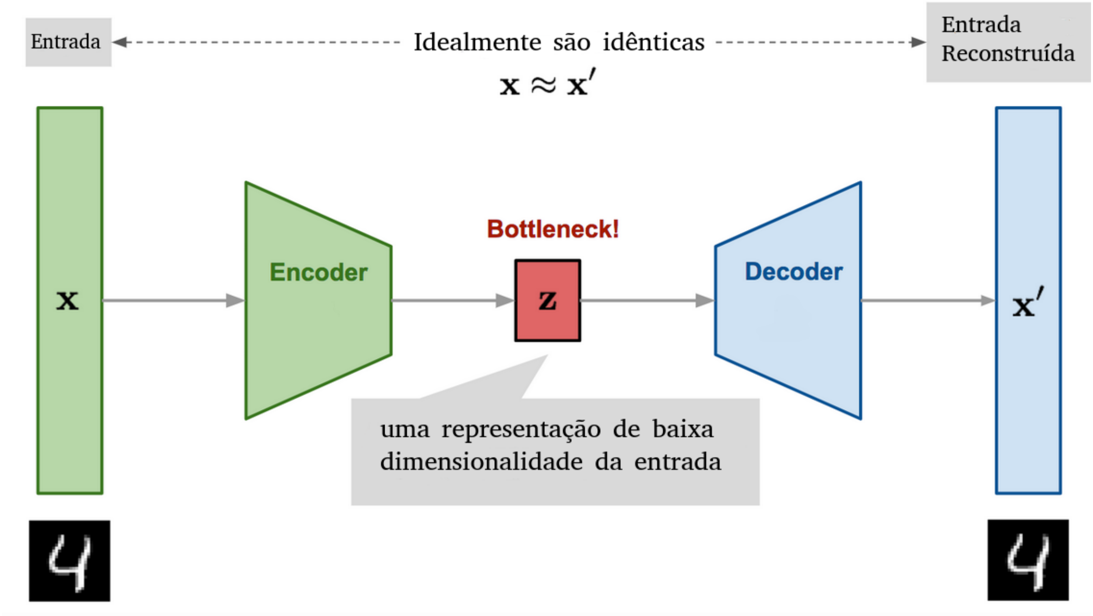

Entrada
Idealmente são idênticas
$\mathbf{x} \approx \mathbf{x}'$
Entrada Reconstruída
x
Encoder
Bottleneck!
z
Decoder
x′
uma representação de baixa dimensionalidade da entrada

Figura 14 – Arquitetura de um *autoencoder*

Fonte: Adaptada de Weng (2018c).

regularizado.

A estrutura geral do VAE pode ser definida de uma maneira semelhante a do *autoencoder* tradicional. Durante o treinamento do VAE, o codificador, também chamado de *probabilistic encoder*, transforma a entrada em uma distribuição no espaço latente. Após, um ponto é amostrado (de forma aleatória) e a partir dessa distribuição, o decodificador transforma esse ponto na entrada. Por fim, o erro da reconstrução é calculado e retropropagado pela rede, ocorrendo dessa forma a atualização dos pesos do modelo. O objetivo de otimização do VAE é o *Evidence Lower Bound* (ELBO) que é composto por dois termos principais. O termo de reconstrução que busca tornar a reconstrução da entrada da rede mais eficiente possível e o termo de regularização que busca regularizar o espaço latente. Desse modo, tenta tornar as distribuições preditas pelo codificador próximas a uma distribuição normal padrão (tendo uma média e desvio padrão próximo de 0 e 1, respectivamente) (KINGMA; WELLING *et al.*, 2019). O termo de regularização é denominado de divergência de *Kullback-Leibler* (KL) (ERVEN; HARREMOS, 2014).

Além disso, como a amostragem é um processo estocástico que não retropropaga o gradiente, é necessário expressar a amostragem de uma forma que seja possível retropropagar o gradiente pela rede. Para tal, foi proposto o truque de reparametrização. Após a aplicação do truque de reparametrização, a Equação 2.13 representa a variável latente (z) amostrada do espaço latente, considerando x como a entrada do modelo (ou uma representação intermediária do codificador, quando o mesmo é composto por várias camadas), μ representa a média, σ representa o desvio padrão e ε representa uma distribuição normal padrão aleatória. Os μ, σ e ε possuem a mesma forma, sendo que o μ e σ podem ser implementados com camadas Perceptron

(KINGMA; WELLING *et al.*, 2019).

$$z(x) = \mu(x) + \sigma(x)\varepsilon \tag{2.13}$$

A Figura 15 apresenta uma representação da arquitetura de um *variational autoencoder*, considerando as definições da Equação 2.13 e x' como a saída predita pelo VAE.

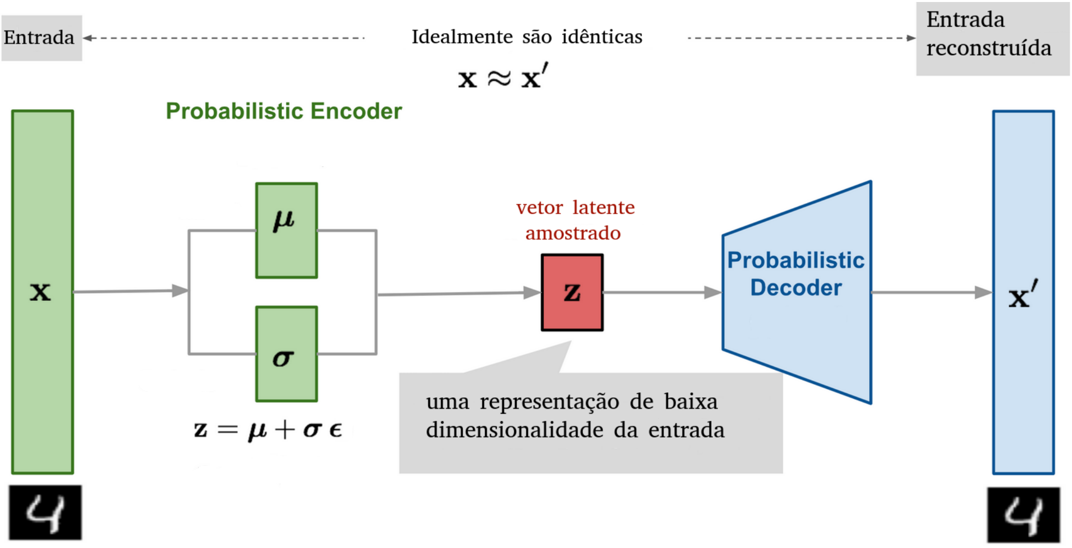


Figura 15 – Arquitetura de um *variational autoencoder*

Fonte: Adaptada de Weng (2018c).

Em várias aplicações os VAE podem ser aplicados e não necessariamente precisam reconstruir a entrada. Um exemplo disso é a aplicação em síntese de fala (ZHANG *et al.*, 2019), em que o VAE é geralmente utilizado para extrair características prosódicas da fala e é treinado como um módulo do modelo de síntese. Nesse caso, a distribuição latente predita pelo VAE é passada em conjunto com a representação textual para o decodificador do modelo de síntese de fala. Além disso, a KL é utilizada para regularizar o espaço latente do VAE e a função de perda do modelo de síntese de fala induz o aprendizado generativo.

Por fim, existem outras variantes do *autoencoder* como *denoising autoencoder* (VINCENT *et al.*, 2008), *sparse autoencoder* (NG *et al.*, 2011), *contractive autoencoder* (RIFAI *et al.*, 2011) e *k-sparse autoencoder* (MAKHZANI; FREY, 2013), entre outros, porém os mesmos não serão abordados nesse trabalho.

2.10 Modelos Flow-based

Na modelagem generativa, os exemplos de treinamento (x) são extraídos de uma distribuição desconhecida $p(x)$. O objetivo de um algoritmo de modelagem generativa é aprender uma $p_{modelo}(x)$ que se aproxime de $p(x)$ o mais próximo possível. Uma forma de aprender uma aproximação de $p(x)$ é utilizar uma função $p_{modelo}(x,\theta)$ parametrizada por θ e procurar o valor dos parâmetros que tornam $p(x)$ e $p_{modelo}(x,\theta)$ mais parecidas possíveis. Uma das abordagens mais populares para a modelagem generativa é a estimativa de máxima (*maximum likelihood estimation*), que é explicitamente a minimização da Kullback-Leibler entre $p(x)$ e $p_{modelo}(x,\theta)$ (GOODFELLOW *et al.*, 2020).

Entretanto, segundo Goodfellow *et al.* (2020), apesar da modelagem de densidade ter funcionado bem para a estatística clássica, com o advento do *deep learning*, as funções $p(x)$ passaram a ser bem mais complexas, especialmente devido à quantidade de parâmetros e profundidade dos modelos. Deste modo, quando utiliza-se uma rede neural profunda para a geração de dados, a função de densidade correspondente pode ser intratável computacionalmente. Para contornar esse problema foram propostas duas abordagens. A primeira é projetar o modelo de forma que a função de densidade seja tratável (FREY; BRENDAN; FREY, 1998). A segunda é projetar um modelo baseado em uma aproximação computacionalmente tratável de uma função de densidade intratável (KINGMA; WELLING, 2013). Os VAEs vistos anteriormente se enquadram nessa categoria. Além disso, surgiram também os modelos generativos implícitos, como as *Generative Adversarial Networks* (GANs) (GOODFELLOW *et al.*, 2014), que evitam todos os problemas de projetar uma função de densidade tratável e aprendem apenas um processo de geração de amostra de uma forma tratável.

As GANs e os VAEs apresentam desempenho impressionante em diversas tarefas desafiadoras (GOODFELLOW *et al.*, 2016). No entanto, segundo Kobyzev, Prince e Brubaker (2020) o fato de nenhum dos dois permitir a avaliação exata da densidade de probabilidade de novos pontos pode limitar sua aplicação na prática. Para uma aproximação de distribuição melhor e mais poderosa, surgiram os *Normalizing Flows* (NF) (DINH; KRUEGER; BENGIO, 2014; REZENDE; MOHAMED, 2015), também referidos como modelos *flow-based* em (PRENGER; VALLE; CATANZARO, 2019; VALLE *et al.*, 2020; KIM *et al.*, 2020).

Os NF foram introduzidos pela primeira vez no aprendizado de máquina no contexto de inferência variacional (REZENDE; MOHAMED, 2015) e estimativa de densidade (DINH; KRUEGER; BENGIO, 2014) e após foram aplicados também em modelagem generativa (ABDELHAMED; BRUBAKER; BROWN, 2019). Nos NF a amostragem e a avaliação da densidade podem ser eficientes e exatas (KOBYZEV; PRINCE; BRUBAKER, 2020). Esses modelos já foram aplicados com sucesso em diferentes tarefas, como geração de imagem (HO *et al.*, 2019), geração de vídeo (KUMAR *et al.*, 2019), síntese de fala (VALLE *et al.*, 2020; KIM *et al.*, 2020), na física (KANWAR *et al.*, 2020), entre outras diversas aplicações.

Um NF consiste na transformação de uma variável aleatória utilizando uma distribuição conhecida (por exemplo, uma distribuição normal) por meio de uma sequência de mapeamentos diferenciáveis e invertíveis (ABDELHAMED; BRUBAKER; BROWN, 2019). Desse modo, o NF transforma uma distribuição simples em uma mais complexa fluindo a mesma por uma cadeia de transformações, substituindo repetidamente a variável pela nova de acordo com o teorema da mudança de variáveis (JEFFREYS; JEFFREYS, 1988; LAX, 1999). Por esse motivo, os modelos *flow-based* conseguem estimar a probabilidade quase ou exata dos dados de entrada (ABDELHAMED; BRUBAKER; BROWN, 2019; KOBYZEV; PRINCE; BRUBAKER, 2020). A Figura 16 apresenta o passo a passo de um NF transformando uma distribuição simples $p_0(z_0)$ em uma distribuição mais complexa $p_K(z_K)$, considerando x como a variável alvo.

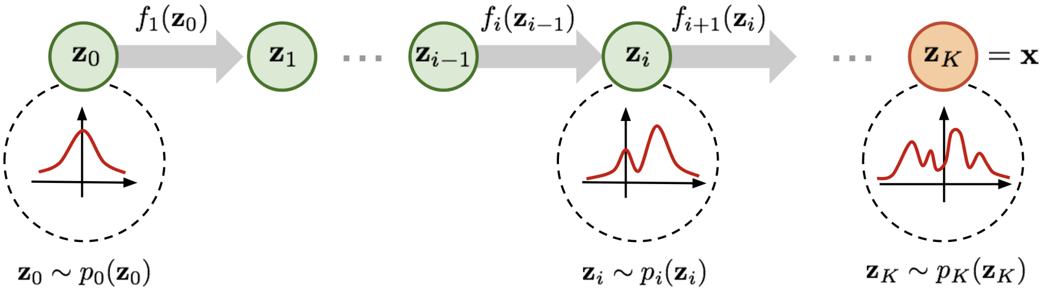

$f_1(\mathbf{z}_0)$
$\mathbf{z}_0$
$\mathbf{z}_1$
$\mathbf{z}_{i-1}$
$f_i(\mathbf{z}_{i-1})$
$\mathbf{z}_i$
$f_{i+1}(\mathbf{z}_i)$
$\mathbf{z}_K = \mathbf{x}$
$\mathbf{z}_0 \sim p_0(\mathbf{z}_0)$
$\mathbf{z}_i \sim p_i(\mathbf{z}_i)$
$\mathbf{z}_K \sim p_K(\mathbf{z}_K)$

Figura 16 – Exemplo de um *Normalizing Flow*

Fonte: Weng (2018b).

Os modelos *flow-based* são treinados com a função de perda *negative log-likelihood* (NLL) (BOSMAN; THIERENS, 2000). Deste modo, o modelo aprende diretamente uma estimativa muito próxima (ou exata) da distribuição $p(x)$ (KIM *et al.*, 2020).

Dado que nos modelos *flow-based* o modelo aprende uma estimativa muito próxima ou exata da distribuição $p(x)$ e o modelo é inversível, utilizando a distribuição predita pelo modelo para uma dada amostra de entrada, podemos obter novamente a entrada aplicando essa distribuição no modelo invertido. Portanto, nos modelos *flow-based* generativos o decodificador é basicamente o codificador invertido. A Figura 17 compara a arquitetura de um modelo *flow-based* generativo com um VAE, onde a arquitetura mais acima representa um VAE e a arquitetura mais abaixo representa o modelo *flow-based*.

Existem diferentes implementações para as funções invertíveis no NF. Por exemplo, nos modelos NICE (DINH; KRUEGER; BENGIO, 2014) e RealNVP (DINH; SOHL-DICKSTEIN; BENGIO, 2016) os autores empilham uma sequência de funções de transformação bijetiva invertíveis. Por outro lado, Kingma e Dhariwal (2018) propuseram o Glow que utiliza convoluções 1x1 invertíveis. Esse trabalho originou vários outros trabalhos em processamento de fala, como o vocoder WaveGlow (PRENGER; VALLE; CATANZARO, 2019), que mostrou ser possível transformar um espectrograma em formato wav com qualidade, reduzindo o tempo de inferência se comparado com os demais modelos no estado da arte da época. Além disso, os modelos de síntese de fala FlowTron (VALLE *et al.*, 2020) e GlowTTS (KIM *et al.*, 2020) apresentaram resultados no estado da arte, recentemente.

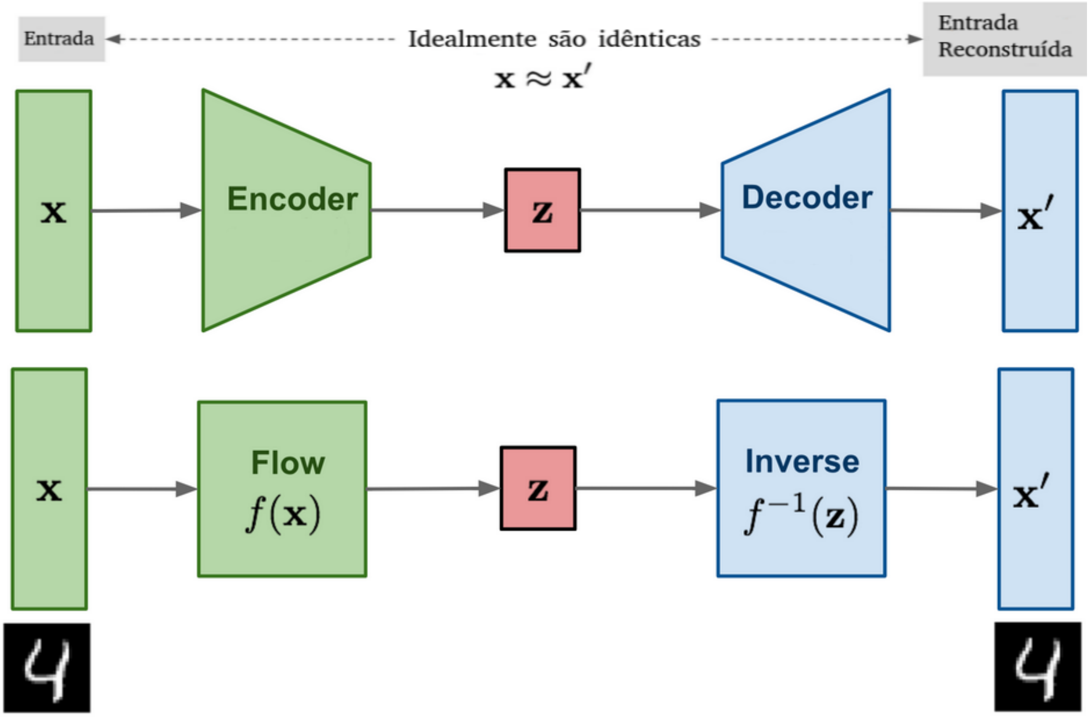


Figura 17 – Comparação de modelo um modelo *flow-based* generativo com um *variational autoencoder*

Fonte: Weng (2018b).

2.11 Pré-processamento de áudio – Espectrogramas, Espectrogramas de Mel e MFCCs

O pré-processamento de áudio desempenha um importante papel, seja no ASR, na verificação de locutores, na síntese de fala e nas diferentes aplicações que envolvam fala. A técnica de extração de atributos *Mel-Frequency Cepstral Coefficients* (MFCCs) (DAVIS; MERMELSTEIN, 1990) foi popular por um longo período, porém, recentemente, os espectrogramas estão se tornando cada vez mais populares e sendo aplicados em abordagens estado da arte nas diferentes tarefas que envolvem fala (VALLE *et al.*, 2020; CHUNG *et al.*, 2020b; LI *et al.*, 2019a).

A primeira etapa para obter-se um espectrograma linear é aplicar um filtro de pré-ênfase no sinal para amplificar as altas frequências. O filtro de pré-ênfase é opcional e utilizado para equilibrar o espectro de frequências, pois as frequências altas geralmente possuem magnitudes menores em comparação com frequências mais baixas, e também melhorar a razão sinal-ruído (RIBANI *et al.*, 2004). Após a pré-ênfase, se faz necessário dividir o sinal em curtos quadros de tempo. Isso é necessário pois a fala é um sinal não estacionário, o que significa que suas propriedades estatísticas não são constantes ao longo do tempo (LOIZOU, 2013). Por esse motivo, em vez de analisar todo o sinal, se faz necessário extrair características de apenas uma pequena janela de fala que pode caracterizar um fone (JURAFSKY; MARTIN, 2014) ou caractere específico e para os quais é feita a suposição de ser estacionário (LOIZOU, 2013; QUINTANILHA, 2017). Portanto, aplica-se uma *Fourier Transform* (FT) sobre esses curtos quadros de tempo, desse modo obtendo-se uma boa aproximação dos contornos de frequência do sinal pela concatenação de quadros adjacentes (GORDILLO, 2013). Segundo Loizou (2013), um espectrograma apresenta a concentração relativa da fala em determinadas frequências em função do tempo.

Um espectrograma linear possui muitas informações e alta dimensionalidade. Portanto, é comum utilizar técnicas que permitem filtrar apenas as frequências que são perceptíveis pelo ouvido humano (LOIZOU, 2013). A mais popular delas é a aplicação da escala de mel ao espectro. A escala de mel tem como objetivo imitar a percepção não-linear do ouvido humano, sendo mais discriminativa nas frequências mais baixas e menos discriminativa nas frequências mais altas (HUANG *et al.*, 2001). Para obtenção de um espectrograma de mel, partindo do espectrograma linear, é necessária a aplicação de filtros triangulares em uma escala de mel no espectro de energia, assim extraindo as bandas de frequência. Após a obtenção das energias filtradas na escala de mel, tira-se o seu logaritmo, obtendo-se o espectrograma de mel (LOIZOU, 2013; QUINTANILHA, 2017).

Os coeficientes dos espectrogramas são muito correlacionados, o que pode vir a causar problemas em algoritmos clássicos de aprendizado de máquina. Para contornar esse problema foram propostos os MFCCs que consistem na aplicação da *Discrete Cosine Transform* (DCT) para descorrelacionar os coeficientes do espectrograma de mel, gerando assim uma representação comprimida do espectrograma de mel (SAHIDULLAH; SAHA, 2012). Apesar da grande popularidade da aplicação dos MFCCs para ASR no passado, com o advento do *deep learning* vários trabalhos mostraram que é possível obter resultados no estado da arte para a tarefa de ASR utilizando espectrogramas (HANNUN *et al.*, 2014; COLLOBERT; PUHRSCH; SYNNAEVE, 2016) ou até mesmo o áudio em formato wav (SCHNEIDER *et al.*, 2019; BAEVSKI *et al.*, 2020; HSU *et al.*, 2021; BAEVSKI *et al.*, 2022), como alternativa aos MFCCs.

2.12 Produção da fala humana

A fala humana é produzida pelo aparelho fonador do falante. Segundo Marquiafável, Bokan e Zavaglia (2014), a fala é gerada por um conjunto de órgãos e os mesmos podem ser subdivididos em três principais grupos: os órgãos respiratórios, como pulmões, brônquios e traqueia, que são responsáveis por produzir as correntes de ar; laringe e pregas vocais, que são os órgãos responsáveis pela produção das vibrações utilizadas na fala; e, por fim, faringe, boca e fossas nasais, que por sua vez são responsáveis pelos diversos sons da fala. A produção do som forma diferentes fluxos de ar, possibilitando a classificação do som em vogais e consoantes. Nas consoantes acontece uma obstrução no fluxo de ar, enquanto que o som passa livremente pelo aparelho fonador do falante na produção das vogais (MARQUIAFÁVEL; BOKAN; ZAVAGLIA, 2014). A abertura das pregas vocais é denominada de glote. A glote, dependendo do seu estado, pode ser classificada em duas fases. A *glottal closed phase* é a fase em que as pregas vocais estão fechadas e portanto nenhum ar flui por meio delas. Por outro lado, a *glottal open phase* é a fase em que as pregas vocais estão abertas. O *pitch period* pode ser definido como o tempo total de duração de um ciclo glotal, e a frequência desse ciclo é chamada de frequência fundamental (do inglês, *fundamental frequency*). A frequência fundamental, também conhecida como F0, geralmente é expressa em Hertz. Os homens geralmente possuem uma frequência fundamental mais baixa que as mulheres (REDFORD, 2015).

2.13 Métricas de avaliação

Nesta seção, são apresentadas as principais métricas utilizadas para a avaliação de modelos de ASR, verificação de locutores e síntese de fala.

Uma das métricas mais populares para avaliação de modelos ASR é a *Word Error Rate* (WER). Considerando que a sequência de palavras preditas pelo sistema ASR está alinhada com uma transcrição de referência, o WER pode ser calculado com a soma da quantidade de palavras substituídas (S), inseridas (I) e deletadas (D) da sequência de palavras predita pelo sistema ASR (ALI; RENALS, 2018). Portanto, considerando que a transcrição de referência tenha N palavras, a WER pode ser calculada da seguinte forma:

$$WER = \frac{S+I+D}{N} \cdot 100 \tag{2.14}$$

Além disso, muitos trabalhos utilizam a métrica *Character Error Rate* (CER) (GUYON *et al.*, 1998) para a avaliação de modelos ASR. O CER é muito similar ao WER entretanto, nessa métrica considera-se o número de substituições, inserções, remoções e a quantidade de caracteres em vez de palavras. O CER é particularmente útil para uma avaliação mais realista do desempenho dos modelos ASR em frases curtas. Por exemplo, se uma frase tiver apenas duas palavras a falta de uma letra em uma dessas palavras irá resultar em um WER de 50%, enquanto o CER vai ser muito menor pois essa métrica considera todos os caracteres em vez das palavras.

Por outro lado, o desempenho dos modelos de verificação de locutores são comumente avaliados pela medida *Equal Error Rate* (EER) (CHENG; WANG, 2004). EER é um algoritmo de sistema de segurança biométrico usado para predeterminar os valores limites da taxa de aceitação falsa e da taxa de rejeição falsa (CHENG; WANG, 2004). Quando ambas as taxas de aceitação falsa e rejeição falsa são iguais esse valor é referenciado como *equal error rate* (WANGKEEREE; BOONKRONG, 2019). Quanto menor o valor do EER, maior a precisão do sistema biométrico (SZTAHÓ; SZASZÁK; BEKE, 2021).

Por fim, os sistemas de síntese de fala são avaliados utilizando a métrica *Mean Opinion Score* (MOS) (RIBEIRO *et al.*, 2011). Para o cálculo da MOS, humanos avaliam os áudios sintetizados pelo sistema de síntese de fala, atribuindo uma nota; a MOS é a média das notas de todos os avaliadores. Seguindo o trabalho de Ribeiro *et al.* (2011), durante o cálculo da MOS os

Tabela 1 – *Mean Opinion Score*

Avaliação	Qualidade	Distorção
5	Excelente	Imperceptível
4	Boa	Apenas perceptível, mas não irritante
3	Razoável	Perceptível e ligeiramente irritante
2	Pobre	Irritante, mas não inutilizável
1	Ruim	Muito irritante e inutilizável

Fonte – Adaptado de (RIBEIRO *et al.*, 2011)

Tabela 2 – *Similarity Mean Opinion Score*

Avaliação	Similaridade
5	Extremamente semelhante
4	Muito semelhante
3	Razoavelmente semelhante
2	Pouco semelhante
1	Nada semelhante

Fonte – Adaptado de (JIA *et al.*, 2018)

avaliadores devem atribuir uma nota para o áudio que varia de 1 a 5, considerando a qualidade e a naturalidade da fala, seguindo os critérios apresentados na Tabela 1.

Além da MOS, os sistemas *zero-shot multi-speaker TTS* são comumente avaliados utilizando a métrica *Similarity Mean Opinion Score* (Sim-MOS) (JIA *et al.*, 2018). Como a MOS, a Sim-MOS varia de 1 a 5 entretanto, o objetivo da mesma é avaliar quão semelhantes são as vozes dos locutores de dois áudios, onde 1 indica que as vozes são nada semelhantes e 5 que são extremamente semelhantes. A Tabela 2 apresenta os critérios utilizados pelos anotadores humanos durante a avaliação.

Além disso, uma medida bastante utilizada para verificar a semelhança da voz de dois áudios é a *Speaker Encoder Cosine Similarity* (SECS). A SECS consiste no cálculo da similaridade do cosseno entre *speaker embeddings* extraídos dos áudios utilizando um sistema de verificação de locutores, também conhecido como *speaker encoders*. A SECS vária de -1 a 1, e um valor maior indica uma maior similaridade.

2.14 Sistemas de Síntese de fala aplicados a aumento de dados para sistemas de ASR

Os avanços na área de síntese de fala e síntese de fala *zero-shot multi-speaker* motivaram trabalhos que exploram o uso de síntese de fala aplicada na realização de aumento de dados para a tarefa de ASR. Essa seção irá introduzir os principais trabalhos que aplicam síntese de fala para aumento de dados no treinamento de modelos ASR.

Li *et al.* (2018) exploraram o uso de síntese de fala para aumento de dados no treinamento de um modelo de ASR. Para isso, como modelo de síntese de fala os autores utilizaram o modelo Tacotron *Global Style Tokens* (GST) (WANG *et al.*, 2018) treinado com 3 locutores do M-AILABS Speech Dataset[1], no inglês americano. Por outro lado, como modelo de ASR utilizaram um modelo baseado no Wav2Letter (COLLOBERT; PUHRSCH; SYNNAEVE, 2016) com algumas alterações em sua arquitetura. Além disso, como *baseline* os autores treinaram um modelo de ASR apenas com fala humana no *dataset* LibriSpeech (PANAYOTOV *et al.*, 2015), utilizando os subconjuntos train-clean-100, train-clean-360 e train-other-500, totalizando aproximadamente 960 horas de fala de 2.338 locutores. Adicionalmente, utilizando o modelo de síntese de fala, os autores geraram uma versão sintetizada dos subconjuntos supracitados, criando um *dataset* artificial. Durante a síntese da fala, como referências de estilo prosódicas para o GST os autores escolheram amostras aleatórias do *dataset* M-AILABS. Para verificar o efeito da síntese de fala no treinamento dos modelos de ASR, os autores combinaram o *dataset* artificial com o *dataset* real. Na maioria dos experimentos os autores utilizaram 50% de fala sintetizada e 50% de fala real, buscando desse modo manter uma mesma quantidade de fala para o *dataset* real, artificial e combinado. Além disso, os autores treinaram um mesmo modelo com os três *datasets*. Para avaliação utilizaram o subconjunto test-clean e test-other do *dataset* LibriSpeech utilizando como métrica de avaliação o WER. Como resultado, no subconjunto test-clean os modelos treinados com o *dataset* real, combinado, e artificial alcançaram, respectivamente, um WER de 5.10, 4.66 e 49.80. Por outro lado, no subconjunto test-other os modelos treinados com os dados real, combinado e artificial alcançaram, respectivamente, um WER de 16.21, 15.47 e 81.78. Portanto, a combinação de fala sintetizada e real apresentou melhorias no WER de 0.44 no test-clean e 0.74 no test-other. Além disso, o desempenho ruim no treinamento do modelo apenas utilizando fala sintetizada foi justificado pelos autores como sendo decorrente da baixa quantidade de variação da fala sintetizada, dado que foram utilizados apenas 3 locutores.

Rosenberg *et al.* (2019) também avaliaram a viabilidade de melhorar o reconhecimento automático de fala utilizando síntese de fala. Como modelo de síntese de fala os autores utilizaram um modelo baseado no Tacotron 2 (SHEN *et al.*, 2018) em modo multi-locutor, utilizando *embeddings* de um *speaker encoder* externo como em Jia *et al.* (2018). Além disso, para permitir manipulação das características da fala os autores integraram ao modelo um VAE como em Hsu *et al.* (2018). Por outro lado, como modelo de ASR os autores utilizaram o modelo *Listen Attend and Spell* (LAS) com atenção aditiva, como em Chiu *et al.* (2018). Para a validação e teste dos modelos de ASR e síntese de fala os autores utilizaram os subconjuntos de teste e validação padrões do *dataset* LibriSpeech. Por outro lado, para treinamento os autores utilizaram as partições train-clean-100, train-clean-360 e train-other-500, totalizando aproximadamente 960 horas de fala de 2.338 locutores. Os autores treinaram o modelo de síntese de fala no *dataset* LibriSpeech e utilizaram esse modelo para criar uma versão artificial do *dataset* LibriSpeech. Durante a construção do *dataset* artificial, para escolher a identidade do locutor para a pronúncia

[1] https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/

de cada uma das sentenças os autores exploraram 3 abordagens distintas. A abordagem que mostrou-se mais efetiva foi a escolha aleatória de um dos locutores do conjunto de treinamento. Com esse *dataset* artificial, os autores fizeram diversos testes variando a quantidade de horas de fala humana. O modelo de ASR treinado apenas sobre a fala artificial alcançou um WER de 32.44 e 66.10, respectivamente, para os subconjuntos test-clean e test-other. Por outro lado, o modelo treinado com todo o *dataset* de fala humana e fala artificial alcançou um WER de 4.58 e 13.78, respectivamente, no test-clean e test-other. Enquanto, o modelo treinado apenas sobre a fala humana alcança 4.77 e 13.89, respectivamente, no test-clean e test-other. Apesar do treinamento utilizando apenas fala artificial resultar em um modelo com um WER alto, a combinação de fala humana e sintetizada se mostrou bem efetiva, diminuindo o WER em 0.19 e 0.11 para os subconjuntos test-clean e test-other. Entretanto, ainda existe uma grande lacuna entre o desempenho de modelos treinados apenas em fala sintetizada e modelos treinados com fala humana. Adicionalmente, os autores exploraram o problema de diversidade lexical, em que o número de sentenças reais é limitado. Para isso, os autores geraram novas frases utilizando um modelo de língua e sintetizaram utilizando o modelo de síntese de fala. Os autores mostram que o aumento de dados lexicais utilizando um modelo de síntese de fala também pode melhorar o WER dos modelos de ASR.

Laptev *et al.* (2020) exploraram o uso de aumento de dados considerando uma menor quantidade de dados disponíveis. Para isso os autores utilizaram um modelo de síntese de fala baseado no GMVAE-Tacotron (HSU *et al.*, 2018) com algumas modificações. Como modelo ASR os autores utilizaram um modelo baseado no LAS disponível no *framework* ESPnet (WATANABE *et al.*, 2018). Como os demais trabalhos supracitados, os autores utilizaram o *dataset* LibriSpeech, utilizando para validação e teste dos modelos os subconjuntos padrões do dataset. Para o treinamento do modelo de síntese de fala os autores utilizaram o subconjunto train-clean-100, que possui aproximadamente 100 horas de 251 locutores. Utilizando o modelo treinado, os autores sintetizaram todas as sentenças do subconjunto train-clean-360 do LibriSpeech, deste modo, construindo uma versão sintetizada do *dataset* train-clean-360 (esse *dataset* foi denominado tts-aug-360). Para comparar a abordagem de aumento de dados utilizando síntese de fala e uma abordagem semissupervisionada, os autores transcreveram o subconjunto train-clean-360 utilizando um modelo ASR treinado no subconjunto train-clean-100, denominando esse novo conjunto como semi-sup-360. Para ter uma referência do resultado do modelo utilizando uma quantidade de fala igual à quantidade adicionada pelos métodos de aumento de dados, os autores treinaram o modelo ASR nos subconjuntos train-clean-100 e train-clean-360. Os autores apresentaram os resultados com e sem a aplicação de um modelo de língua na correção da saída do modelo ASR, porém aqui serão apresentados apenas os resultados obtidos com a utilização do modelo de língua. O modelo ASR treinado com os subconjuntos train-clean-100 e train-clean-360 alcançou um WER de 3.5 e 9.1, respectivamente, nos subconjuntos test-clean e test-other. Por outro lado, o modelo treinado com os subconjuntos train-clean-100 e semi-sup-360 alcançou 5.2 e 13.0 nos subconjuntos test-clean e test-other. Por fim, o modelo treinado com os subconjuntos train-clean-100 e tts-aug-360 alcançou um WER de 4.3 e 13.5 no test-clean e test-other. Portanto, o aprendizado semissupervisionado comparado com o aumento de dados utilizando síntese de fala alcançou um desempenho inferior em 0.9 no test-clean, por outro lado, conseguiu um desempenho superior em 0.5 no test-other. Segundo os autores, o conjunto test-other é um conjunto mais desafiador, que contém diferentes sotaques e uma qualidade de áudio inferior ao test-clean portanto, a variabilidade maior no uso de áudios reais ajuda o aprendizado semissupervisionado a obter um melhor resultado nesse conjunto. Por fim, o modelo treinado nos subconjunto train-clean-100 e train-clean-360 se comparado ao modelo treinado com o aumento de dados alcançou um desempenho superior em 0.8 e 4.4, respectivamente, no test-clean e test-other. Portanto, o aumento de dados utilizando fala sintetizada alcança um resultado próximo ao uso de fala humana transcrita. Além disso, os autores investigaram se o uso de aumento de dados com frases já existentes no conjunto de treinamento pode melhorar o resultado do modelo ASR. Para isso treinaram um modelo ASR utilizando os subconjuntos train-clean-100, train-clean-360 e tts-aug-360. Esse modelo alcançou um WER de 3.2 e 9.1, respectivamente, no test-clean e test-other. Deste modo, no test-clean alcançou um desempenho 0.3 superior ao modelo treinado apenas com fala humana. Portanto, o aumento de dados utilizando síntese de fala pode melhorar o resultado do modelo de ASR até mesmo quando utiliza sentenças já presentes no conjunto de treinamento. Apesar dos autores mostrarem resultados muito promissores, eles apontam que a principal limitação do trabalho é a quantidade e qualidade de dados necessária para treinamento do modelo de síntese de fala. Línguas com poucos recursos disponíveis provavelmente não terão dados suficientes e adequados para se construir um modelo de síntese de fala multi-locutores funcional, especialmente por essa metodologia necessitar de dados com pouco ou nenhum ruído de fundo e uma quantidade de locutores suficiente.

2.15 Síntese de fala multilíngue

Os recentes avanços em síntese de fala motivaram pesquisadores a projetar modelos que podem aprender mais de um idioma ao mesmo tempo. Alguns desses modelos são particularmente interessantes, pois permitem a realização de *code switching*. *Code switching* consiste em alterar o idioma durante a pronúncia de uma frase, sendo possível pronunciar estrangeirismos de maneira eficiente na língua alvo (NEKVINDA; DUŠEK, 2020). No português brasileiro essa características se torna cada vez mais indispensável em um modelo de síntese de fala dada a grande quantidade de estrangeirismos utilizados no dia a dia.

Cao *et al.* (2019) foi um dos primeiros trabalhos que exploraram síntese de fala multilíngue utilizando um modelo de ponta a ponta. Para isso os autores adaptaram o modelo Tacotron 2. Os autores exploraram duas abordagens distintas. Na primeira, concatenaram *language embeddings* que são aprendidos durante o treinamento do modelo em várias partes do *encoder* do modelo Tacotron. Na segunda abordagem eles utilizaram dois *encoders* diferentes, um para cada idioma. Além disso, para controlar o timbre da saída do modelo adicionaram *speaker embeddings* que também são aprendidos durante o treinamento. Os autores treinaram ambas as abordagens com um *dataset* composto por fala de duas locutoras — uma no idioma inglês americano e outra no mandarim. Os experimentos mostraram a eficácia dos dois sistemas propostos em termos de qualidade e similaridade do locutor da fala gerada. Além disso, mostraram que é possivel fazer *code switching* entre os dois idiomas.

Zhang *et al.* (2019) exploraram a síntese de fala multilíngue também utilizando o modelo Tacotron 2. Entretanto, para transformar o modelo em multilíngue os autores adicionaram *speaker embeddings* e *language embeddings* que são aprendidos durante o treinamento do modelo e concatenaram os mesmos na entrada do *decoder*. Além disso, adicionaram um classificador de locutores treinado adversariamente, buscando remover informações específicas do locutor do *encoder* do modelo, permitindo deste modo o *code switching*. Por fim, buscando permitir a manipulação da fala, adicionaram um VAE residual, como em Hsu *et al.* (2018). Portanto, em vez de adicionar *language embeddings* no *encoder* do modelo ou treinar separadamente um *encoder* para cada idioma, similar ao utilizado por Cao *et al.* (2019), os autores treinaram o *encoder* padrão do modelo Tacotron e deixaram a tarefa de mudança do idioma para o decoder. Deste modo, o *encoder* compartilha informações de todos os idiomas. Os autores treinaram o modelo em um *dataset* de alta qualidade composto por 385 horas no idioma inglês de 84 locutores profissionais com diferentes sotaques, 97 horas no idioma espanhol de 3 locutoras e 68 horas de mandarim de 5 locutores. O modelo final consegue sintetizar fala de alta qualidade em três idiomas. Além disso, o modelo aprendeu a pronunciar estrangeirismos com controle moderado de sotaque, deste modo, permitindo *code switching* entre os três idiomas.

Nekvinda e Dušek (2020) uniram as abordagens propostas por Cao *et al.* (2019) e Zhang *et al.* (2019). Os autores compararam ambas as abordagens em um mesmo *dataset* propondo melhorias. Para realizar uma comparação justa, os autores removeram o VAE residual do modelo de Zhang *et al.* (2019). Além disso, os autores propuseram uma nova abordagem que segue uma abordagem de meta-aprendizado de geração de parâmetros contextuais proposta por Platanios *et al.* (2018). Os autores utilizaram toda a estrutura do modelo de Zhang *et al.* (2019), exceto o VAE residual e o encoder. Em vez de utilizar um *encoder* baseado em recorrência para cada idioma como em Cao *et al.* (2019) ou utilizar um único *encoder* para todos os idiomas os autores utilizaram vários encoders totalmente convolucionais baseados no *encoder* do modelo DCTTS (TACHIBANA; UENOYAMA; AIHARA, 2018). Além disso, para permitir o compartilhamento de conhecimento entre idiomas, os parâmetros dos encoders são gerados usando uma rede totalmente conectada separada que é condicionada a *language embeddings*. O gerador de parâmetros é composto de vários geradores específicos, e cada gerador recebe o identificador de um idioma de entrada e produz parâmetros para uma camada do *encoder* convolucional para esse idioma. Segundo os autores, os geradores permitem um compartilhamento de parâmetros multilíngue controlável, pois o seu poder computacional impede a geração de parâmetros altamente específicos do idioma. Para comparar a abordagem proposta com as abordagens de Cao *et al.* (2019) e Zhang *et al.* (2019), os autores treinaram as três abordagens em um *dataset* composto por 10 idiomas construído a partir do *dataset* CSS10 (PARK; MULC, 2019) e de uma porção do *dataset* Common Voice (ARDILA *et al.*, 2019). Os autores mostraram que o modelo proposto por eles alcançou um resultado melhor na *code switching* que os demais em termos qualidade, naturalidade e precisão de pronúncia.

Por fim, Li *et al.* (2021) exploraram a síntese de fala multilíngue buscando a diminuição do custo computacional dos modelos. Os autores propuseram o modelo Light-TTS, que utiliza Dynamic convolution (WU *et al.*, 2018), diminuindo significativamente o custo computacional do modelo de síntese de fala. Os autores adicionaram ao modelo *language embeddings* e *speaker embeddings*. Os *language embeddings* são aprendidos durante o treinamento, enquanto que os *speaker embeddings* são extraídos do modelo de verificação de locutores x-vector (SNYDER *et al.*, 2018). Além disso, para permitir a produção de fala expressiva os autores utilizaram como entrada do modelo as características *pitch* e *energy*. Por fim, os autores treinaram seus modelos utilizando 30 horas de 130 locutores no inglês e 30 horas de 50 locutores no chinês. Após o treinamento, compararam seus modelos com duas *baselines* propostas: uma baseada no modelo Tacotron 2 e outra no FastSpeech. Apesar disso, os autores não compararam seus resultados com nenhum dos modelos supracitados. Por fim, o modelo proposto consegue gerar fala com qualidade similar aos modelos *baselines*, porém tendo um custo computacional muito menor.

CAPÍTULO

3

DEEP LEARNING APPROACHES FOR SPEECH SYNTHESIS AND SPEAKER VERIFICATION

Título:	**Deep Learning approaches for Speech Synthesis and Speaker Verification**
Autores:	**Edresson Casanova, Christopher Shulby e Sandra Aluisio**
Ano:	**2021**
Workshop:	**Acoustic Communication: An Interdisciplinary Approach (November 19-20, 2020)**
Livro:	**Portal de Livros Abertos da USP (August 31, 2021): http://www.livrosabertos.sibi.usp.br/portaldelivrosUSP/catalog/book/647**
Situação:	**Publicado**

Contribuições relevantes para a tese:

- Esse capítulo apresenta conceitos básicos e o estado da arte em síntese de fala e verificação de locutores.

Deep Learning approaches for Speech Synthesis and Speaker Verification

Edresson Casanova[a], Christopher Shulby[b] and Sandra Maria Aluisio[a]

[a]Institute of Mathematics and Computer Science, University of São Paulo, São Carlos - SP, Brazil
[b]DefinedCrowd Corp., Seattle - WA, USA

ARTICLE HISTORY
Compiled January 3, 2022

ABSTRACT
Speech synthesis is the artificial production of human speech, which can be used in applications like music generation, navigation systems and accessibility for visually-impaired people. As for the speaker recognition task, we can define it as the process of recognizing the speaker of a speech segment by processing speech signals, which can be broadly classified as speaker identification and verification. This article summarizes the Deep Learning practices applied in the field of speech synthesis and speaker verification. Speech synthesis and speaker verification have been widely explored in speech technology applications, especially due to the popularity of virtual assistants. Much research work has been done and significant progress has been made in the last 5-6 years. As Deep Learning techniques advance in most fields of machine learning, older state-of-the-art methods have also being replaced by methods based on Deep Learning in both speech synthesis and speaker verification areas. Therefore, apparently Deep Learning has become the next generation solution for the synthesis and verification of speakers.

KEYWORDS
Speech Technologies; Speech Synthesis; Speaker Verification; Deep Learning approaches

1. Introduction

Speech synthesis systems, also known as Text-To-Speech (TTS), have received a lot of attention in recent years due to the popularization of virtual assistants, such as Amazon Echo (Purington et al. 2017), Google Home (Dempsey 2017) and Apple Siri (Gruber 2009). However, according to Tachibana et al. (2017) traditional Speech Synthesis systems are not easy to develop, as these systems are typically composed of many specific modules, such as, a text analyzer, a grapheme-to-phoneme converter, a duration estimator, an F0 generator, a spectrum generator and a vocoder. Figure 1 presents the main components of a traditional speech synthesis system. In summary, given an input text, the text analyzer module converts dates, currency symbols, abbreviations, acronyms and numbers into their standard formats to be pronounced or read by the system, i.e. carries out the text normalization and tackles problems like homographs, then with the normalized text the phonetic analyzer converts the grapheme into phonemes. In turn, the duration estimator estimates the duration of

CONTACT Edresson Casanova Email: edresson@usp.br

each phoneme. The acoustic model is used to generate acoustic characteristics such as F0 and a spectral envelope that corresponds to linguistic characteristics. Finally, the vocoder converts the spectrum into a waveform (Ze et al. 2013).

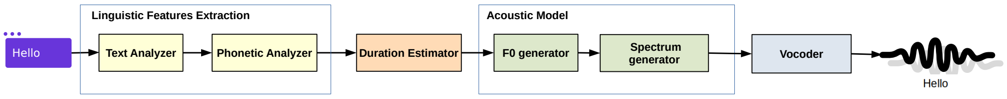


Figure 1. The main components of a traditional speech synthesis system.

The advent of Deep Learning (Goodfellow et al. 2016) has made it possible to integrate all processing steps into a single model and connect them directly from the input text to the synthesized audio output, which is known as end-to-end learning. Although neural models are sometimes criticized as being difficult to interpret, several end-to-end trained speech synthesis systems such as (Kyle et al. 2017; Y. Wang et al. 2017; Shen et al. 2018; Tachibana et al. 2017; Ping et al. 2018; Kim et al. 2020; Valle et al. 2020) have been able to estimate spectrograms from text entries with promising performances.

Due to the sequential characteristic of text and audio data, the recurring units were the standard building blocks for speech synthesis, as in Tacotron 1 and 2 (Y. Wang et al. 2017; Shen et al. 2018). In addition, the convolutional layers showed good performance while reducing computational costs as shown in the DeepVoice 3 (Ping et al. 2018) and Deep Convolutional Text To Speech (DCTTS) (Tachibana et al. 2017) models. On the other hand, with the recent popularization of Transformers (Vaswani et al. 2017) some synthesis models based on transformers have emerged, such as the work proposed by Li et al. (2019), which reached a performance similar to Tacotron 2 (Shen et al. 2018), being trained 4.25 times faster than Tacotron 2. And, finally, the flow-based models (Kingma et al. 2016; Hoogeboom et al. 2019; Durkan et al. 2019) received attention in the speech synthesis area, where the Flowtron (Valle et al. 2020) model surpassed the results reported by Tacotron 2 for enabling the manipulation of the latent space, allowing to change characteristics such as speech speed and prosody. On the other hand, Kim et al. (2020) proposed GlowTTS, which achieved a similar performance as the Tacotron 2 synthesizing speech 15.7 times faster.

The advent of Deep Learning has also enabled great advances for the task of speaker recognition. Speaker Recognition can be divided into three different subtasks: Speaker Verification (SV), Speaker Identification and Speaker Diarization. The objective of SV is to say if two distinct audios contain the voice of the same speaker. On the other hand, the speaker identification seeks to find which speaker produced the voice of the audio file. Finally, the Speaker Diarization splits an input audio stream into homogeneous segments according to the speaker identity.

In this work we will only deal with the Speaker Verification since it can be used in both of the other tasks cited above (Sztahó et al. 2019; Casanova, Candido Junior, Shulby, et al. 2020). Currently, Speaker Verification state-of-the-art (SOTA) systems (J. Wang et al. 2019; Deng et al. 2019; J. S. Chung et al. 2020; Casanova, Candido Junior, Shulby, et al. 2020) allow the identification of new speakers without the need for retraining the model. This feature is very useful for different applications, such as meeting scanners, telephone-banking systems (Bowater & Porter 2001) and automatic question answering (Ferrucci et al. 2010).

Table 1. Speech Synthesis datasets

Corpus	**Hours (~)**	**Total Speakers (~)**
LibriTTS (Zen et al. 2019)	586	2,456
M-AILAB[1]	75	2
VCTK (Veaux et al. 2016)	44	109
LJ Speech (Ito 2017)	24	1
TTS-Portuguese Corpus (Casanova, Candido Junior, de Oliveira, et al. 2020)	10.5	1

The objective of this study is to review the SOTA methods using Deep Learning that are applied in the speech synthesis area, focusing on Sequence-to-Sequence (seq2seq) models and speaker verification tasks. This work is organized as follows. The Section 2 presents the main datasets employed in speech synthesis and speaker verification tasks. Section 3 presents the main approaches based on Deep Learning for the speech synthesis task. Section 4 in turn presents the main approaches based on Deep Learning for the task of Speaker Verification. And finally, Section 5 presents the final conclusions and reflections.

2. Speech datasets

As with many tasks related to machine learning, the issue of the dataset used is fundamental. The methods developed can be evaluated and compared only if the same test circumstances are used. It is difficult to say that an approach performs better if it is evaluated on a different dataset (or corpus) (Sztahó et al. 2019). There are some datasets created and used for speaker recognition and speech synthesis. The Section 2.1 presents the most commonly used datasets for speech synthesis in the English language and also presents the unique dataset publicly available for Brazilian Portuguese. In turn, the Section 2.2 presents the main datasets used in the training and evaluation of speaker recognition models.

2.1. Speech Synthesis datasets

For the task of speech synthesis, high quality datasets recorded in controlled environments are required. The purpose of speech synthesis is to synthesize high quality voice, therefore if the training dataset contains noise, the model can synthesize noise and this is not desired. The most used datasets for training single-speaker speech synthesis models is the LJ Speech (Ito 2017) dataset, which consists of 24 hours of speech by an English-language speaker. On the other hand, for multi-speaker synthesis the LibriTTS (Zen et al. 2019) and VCTK (Veaux et al. 2016) datasets are the most used ones. Although the most popular datasets are for the English language, other languages also have open datasets. For Portuguese, for example, the unique publicly available dataset is the TTS-Portuguese Corpus (Casanova, Candido Junior, de Oliveira, et al. 2020). Table 1 shows the approximate number of hours and total number of speakers of the main publicly available datasets for speech synthesis in English and the unique dataset available for Portuguese.

Table 2. Speaker Verification datasets

Corpus	Hours (~)	Total Speakers (~)
LibriSpeech (Panayotov et al. 2015)	986	2,484
Common Voice (Ardila et al. 2019)	2,508	58,250
TED-LIUM V3 (Hernandez et al. 2018)	452	2,028
VoxCeleb (J. S. Chung et al. 2018)	2,000	6,112

2.2. Speaker Verification datasets

For the SV task, the datasets created for the development of Automatic Speech Recognition (ASR) systems are commonly used due to their characteristics. Unlike speech synthesis datasets, ASR datasets generally have many speakers and few samples for each speaker; this feature is desired since for Speaker Verification we want to have as many speakers as possible during model training (Sztahó et al. 2019). Therefore, the datasets that are built for ASR models can be used for training and evaluation of SV models. However, there are datasets made specially for Speaker Verification. For example, VoxCeleb 2 (J. S. Chung et al. 2018) is currently the largest dataset built for SV. VoxCeleb 2 consists of samples from more than 6,000 speakers downloaded from Youtube. Table 2 shows the approximate number of hours and total number of speakers of the main publicly available datasets for ASR and additionally provides information about the VoxCeleb 2 dataset.

3. Sequence-to-Sequence Speech Synthesis Approaches

With the advent of Deep Learning, speech synthesis systems have evolved greatly and are still being studied intensively. Models based on Recurrent Neural Networks such as Tacotron (Y. Wang et al. 2017), Tacotron 2 (Shen et al. 2018), Deep Voice 1 (Arik et al. 2017) and Deep Voice 2 (Arık et al. 2017) have gained prominence, but as these models use recurring layers they have high computational costs. This led to the development of fully convolutional models, such as DCTTS (Tachibana et al. 2017) and Deep Voice 3 (Ping et al. 2018), which sought to reduce the computational cost while maintaining the good quality of the synthesis. On the other hand, more recently with the popularization of the Transformers, new models based on Transformers (Li et al. 2019; Kim et al. 2020) have emerged, due to the parallelization of this architecture, the based models managed to achieve similar results to the recurrent architectures with a lower computing cost. And finally, the flow-based models (Kingma et al. 2016; Hoogeboom et al. 2019; Durkan et al. 2019) received attention in the synthesis area, allowing the training of simpler models and with a reduced computational cost. For example, the GlowTTS (Kim et al. 2020) model achieved similar quality to the recurrent Tacotron 2 model, but it can synthesize speech 15.7 times faster.

The speech synthesis models are trained by receiving a text as input and a spectrogram as an expected output that represents the speech of the respective text input. The model must learn to generate a spectrogram given the input text, after that, the spectrogram is transformed into a waveform using a vocoder. Neural vocoders have a higher quality speech synthesis, while phase reconstruction methods such as Griffin-Lim (GLA) (Griffin & Lim 1984) and RTISI-LA (Real-Time Iterative Spectrogram Inversion with Look-Ahead) (Zhu et al. 2007) are based on the Short Fast Fourier Transform (SFFT) (Sorensen & Burrus 1988) redundancy and have a higher synthesis speed with reduced quality. Figure 2 presents a general flow diagram of a TTS system based on Deep Learning. Briefly, given an input text, it is passed to the TTS model which returns a spectrogram. Finally, this spectrogram is converted into a waveform by the vocoder.

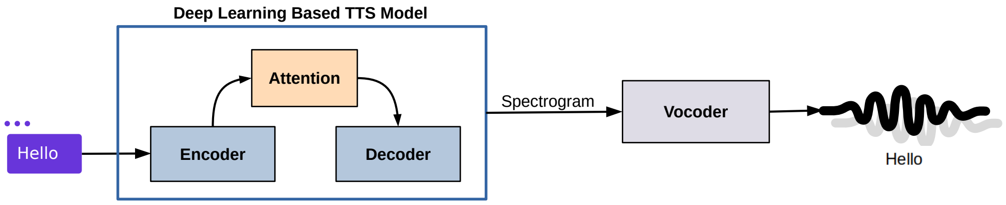


Figure 2. General flow diagram of a TTS system based on Deep Learning.

The most popular neural vocoders today are Wavenet (Tamamori et al. 2017), WaveRNN (Kalchbrenner et al. 2018), Waveglow (Prenger et al. 2019), GAN-TTS (Bińkowski et al. 2019), Melgan (Kumar et al. 2019) and more recently WaveGrad (Chen et al. 2020). Each of these vocoders has its advantages; some focus on higher quality and others on faster synthesis. In this work, we will not detail vocoders, but it plays a very important role in speech synthesis, converting a spectrogram into a waveform. In this work, we will only focus on models that convert text into spectrograms.

For the training of speech synthesis models, a large amount of data is required, as mentioned above, for the English language the most popular single speaker dataset for speech synthesis is called LJ Speech (Ito 2017) and has 24 hours of speech on the other hand, in Brazilian Portuguese the unique available dataset is called TTS-Portuguese Corpus (Casanova, Candido Junior, de Oliveira, et al. 2020) and has 10 hours of speech. The speech synthesis models are subjectively evaluated using the Mean Opinion Score (MOS) measure (Ribeiro et al. 2011). Ribeiro et al. (2011) proposed a methodology for calculating MOS in speech synthesis and the vast majority of works follow this methodology. To calculate the MOS, the evaluators are asked to assess the naturalness of the statements generated on a five-point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). After that each subject evaluates the audios, the MOS is calculated an average among the opinion of each subject; this average represents the MOS value.

Tacotron 1 (Y. Wang et al. 2017) was one of the first models of speech synthesis to use only neural networks to transform text into a spectrogram. The authors proposed the use of a single deep neural network trained from end to end. Tacotron 1 includes an encoder and a decoder. The model uses an attention mechanism (Bahdanau et al. 2014) and also includes a post-processing module. This model uses convolutional filters, skipping connections (Srivastava et al. 2015) and Gated Recurrent Units (GRUs) (J. Chung et al. 2014) neurons. Tacotron also uses the Griffin-Lim (Griffin & Lim 1984) algorithm to convert the spectrogram STFT to the waveform. In parallel, the Deep Voice 1 (Arik et al. 2017) model also appeared, which used several neural submodels to synthesize speech into text. Then the Deep Voice 2 (Arık et al. 2017) model was proposed. This model is based on Deep Voice 1, however the authors proposed some better ones to overcome the results obtained by Tacotron 1. In addition, the authors proposed improvements in Tacotron 1 and changed the Griffin-Lim vocoder in favor of the WaveNet neural vocoder increasing the quality of synthesized speech.

On the other hand, in order to increase the quality Shen et al. (2018), they proposed Tacotron 2. Tacotron 2 is an improvement on the Tacotron 1 model, to which the authors simplified the architecture and combined this new model with a modified version of the WaveNet (Tamamori et al. 2017) vocoder. Tacotron 2 is composed of a recurrent network of sequence prediction features that maps the incorporation of characters to Mel spectrograms, followed by a modified WaveNet model acting as a vocoder to synthesize waveforms in the time domain from these spectrograms. They also demonstrated that the use of Mel spectrograms as a conditioning input for WaveNet, instead of linguistic characteristics, which allows for a significant reduction in the size of the WaveNet architecture, and consequently a faster speech synthesis.

Furthermore, with the popularization of Transformers (Vaswani et al. 2017) in the Natural Language Processing area and the use of many language models such as BERT (Devlin et al. 2018), some synthesis models based on transformers have emerged. We can cite the work proposed by (Li et al. 2019) which achieved a similar quality to Tacotron 2 (Shen et al. 2018) being trained 4.25 times faster than Tacotron 2.

Finally, more recently the flow-based models (Kingma et al. 2016; Hoogeboom et al. 2019; Durkan et al. 2019) received attention in the area of synthesis. Valle et al. (2020) proposed the Flowtron model, which reformulates from Tacotron 2 to provide high-quality and expressive Mel spectrogram synthesis. Flowtron is optimized to maximize the likelihood of training data, which makes training simple and more stable. It allows the manipulation of many aspects of speech synthesis, such as pitch, tone, speech rate, cadence and accent. It achieved MOS scores slightly higher than the Tacotron 2 model and also allows for speech manipulation. On the other hand, Kim et al. (2020) proposed GlowTTS, which achieved a similar quality to Tacotron 2 synthesizing speech 15.7 times faster, GlowTTS uses transformers in its architecture and also allows one to manipulate the speech speed. Both Flowtron and GlowTTS models use the WaveGlow neural vocoder.

4. Speaker Verification approaches

In the last decade, the area of speaker recognition has undergone major changes. In the past, speaker identification models could only identify speakers seen during training and required a reasonable amount of data of a speaker to be able to learn to identify that speaker. Currently, speaker recognition models are able to identify speakers not seen in training using just a few seconds of the voice of the speaker; this is known as the open-set scenario. This advance was possible due to the evolution of the machine learning area and the introduction of new cost functions applied to the training of these models.

Current speaker verification methods are trained using acoustic features, such as Mel-frequency cepstral coefficients (MFCCs) (Davis & Mermelstein 1990) or Mel spectrograms, as inputs and use speaker IDs to calculate the loss. The models aim to learn a representation (speaker embedding), which is a vector of fixed size, to which the distance of the vectors of two different speakers is the greatest possible, while the distance of vectors of two samples of the same speaker are as close as possible. After training, the distance between these embeddings is usually calculated, thus being able to identify the speakers. The performance of SV systems is commonly evaluated by the Equal Error Rate (EER) measure (Cheng & Wang 2004). EER is a biometric security system algorithm used to predetermine the threshold values for its false acceptance index and its false rejection rate (Cheng & Wang 2004). The EER measure indicates that the proportion of false acceptances is equal to the proportion of false rejections, and the lower the value of the EER, the greater the precision of the biometric system

(Sztahó et al. 2019).

An SV system can be evaluated in two scenarios. The Closed-set scenario, where samples of speakers seen in the training of the SV model are used, therefore, the model knows these speakers. In the Open-set scenario, where speaker samples never seen in the training of the model are used, the model does not know these speakers. The models usually report only EER results for the Open-set scenario, since the goal of SV systems is to learn to differentiate speakers never seen in training, eliminating the need to retrain the neural model (Casanova, Candido Junior, Shulby, et al. 2020).

The first studies to use deep neural networks in speaker recognition in an open set scenario used speaker embeddings learned using the Softmax loss. Although the Softmax classifier can learn different embeddings for different speakers (Snyder et al. 2017, 2018), it proved not to be discriminatory enough (J. S. Chung et al. 2020). To overcome this problem, the models trained with Softmax were combined with back ends built in the Probabilistic Linear Discriminant Analysis (PLDA) (Ioffe 2006) to generate scoring functions (Ramoji et al. 2020; Snyder et al. 2018). On the other hand, Liu et al. (2017) proposed Softmax Angular where the cosine similarity is used as logit input for the Softmax layer, showing its superiority over the use of Softmax only. Subsequently, F. Wang et al. (2018) proposed the use of Additive Margins in Softmax (AM-Softmax) to increase inter-class variance by introducing a cosine margin penalty in the target logit. However, according to J. S. Chung et al. (2020) training with AM-Softmax and AAM-Softmax (Deng et al. 2019) proved to be a challenge, as they are sensitive to scale and margin value in the loss function.

The use of contrastive loss (Chopra et al. 2005) and triple loss (Schroff et al. 2015; Bredin 2017) has also achieved promising results in speaker recognition, but these methods require a careful choice of pairs or triplets, which costs time and can interfere with performance (J. S. Chung et al. 2020).

J. Wang et al. (2019) proposed the use of prototypical networks (Snell et al. 2017) in the recognition of speakers. Prototypical networks seek to learn a metric space in which the classification of open sets of speakers can be performed by calculating distances for prototypical representations of each class. Generalized end-to-end loss (GE2E) (Wan et al. 2018) and Angular Prototypical (J. S. Chung et al. 2020) follow the same principle and achieved SOTA results in speaker recognition recently.

J. S. Chung et al. (2020) proposed the comparison of different loss functions mentioned above in the training of two convolutional models proposed by the authors. The authors showed that the Prototypical Angular loss function has a superior performance than the others, thus showing that it is more suitable for training models for SV.

Finally, Casanova, Candido Junior, Shulby, et al. (2020) proposed a new training approach that consists of reconstructing the 1-second pronunciation of the phoneme /a/ constant in the voice of the speakers. After training, the model is able to approximate the pronunciation of the phoneme /a/ in the voice of any speaker and an embedding of this reconstruction is extracted from an intermediate layer of the neural network. As the reconstruction of the phoneme /a/ from the same speaker is always closer to his own than to others, the model is applied in open-set scenarios. In addition, the method surpassed a model trained in a 500x larger dataset with the GE2E loss function. Moreover, the method surpassed the result of the best model proposed by J. S. Chung et al. (2020) and trained with the Angular Prototypical loss function in one of the four datasets used to compare the models. Therefore, it is a method that requires less data to achieve competitive results.

5. Conclusions

In this article, we aimed to list the main Deep Learning approaches applied in the field of Speech Synthesis and Speaker Verification. In the era of Deep Learning, as in most tasks involving machine learning, significant increases in performance compared to classic/traditional methods have been seen. As Deep Learning techniques advance in most fields of machine learning, older, state-of-the-art methods are also being replaced by methods based using Deep Learning in both speech synthesis and Speaker Verification. Therefore, Deep Learning has apparently become the next generation solution for speech synthesis and speaker verification (Sztahó et al. 2019). In some cases, Deep Learning opened up new research fronts, allowing us to meet demands that were not previously met. In addition, speaker verification and speech synthesis systems are still evolving. In the Speech Synthesis field, the current goal is to reduce the computational cost of the models and improve the speech manipulation mechanisms seeking the possibility of synthesizing more expressive speech (Valle et al. 2020; Kim et al. 2020). On the other hand, in Speaker Verification, researchers still seek to advance the current results and focus more on new training methods for modeling (J. S. Chung et al. 2020; Casanova, Candido Junior, Shulby, et al. 2020).

References

Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., ... Weber, G. (2019). Common voice: A massively-multilingual speech corpus. *arXiv preprint arXiv:1912.06670*.

Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., ... others (2017). Deep voice: Real-time neural text-to-speech. *arXiv preprint arXiv:1702.07825*.

Arık, S. O., Diamos, G., Gibiansky, A., Miller, J., Peng, K., Ping, W., ... Zhou, Y. (2017). Deep voice 2: Multi-speaker neural text-to-speech. *arXiv preprint arXiv:1705.08947*.

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Bińkowski, M., Donahue, J., Dieleman, S., Clark, A., Elsen, E., Casagrande, N., ... Simonyan, K. (2019). High fidelity speech synthesis with adversarial networks. *arXiv preprint arXiv:1909.11646*.

Bowater, R. J., & Porter, L. L. (2001, August 21). *Voice recognition of telephone conversations.* Google Patents. (US Patent 6,278,772)

Bredin, H. (2017). Tristounet: triplet loss for speaker turn embedding. In *Acoustics, speech and signal processing (ICASSP), 2017 ieee international conference on* (pp. 5430–5434).

Casanova, E., Candido Junior, A., de Oliveira, F. S., Shulby, C., Teixeira, J. P., Ponti, M. A., & Aluisio, S. M. (2020). End-to-end speech synthesis applied to Brazilian Portuguese. *arXiv preprint arXiv:2005.05144*.

Casanova, E., Candido Junior, A., Shulby, C., da Silva, H. P., Cordeiro, A. F., Guedes, V. d. O., & Aluisio, S. M. (2020). Speech2phone: A multilingual and text independent speaker identification model. *arXiv preprint arXiv:2002.11213*.

Chen, N., Zhang, Y., Zen, H., Weiss, R. J., Norouzi, M., & Chan, W. (2020). Wavegrad: Estimating gradients for waveform generation. *arXiv preprint arXiv:2009.00713*.

Cheng, J.-M., & Wang, H.-C. (2004). A method of estimating the equal error rate for automatic speaker verification. In *2004 international symposium on Chinese spoken language processing* (pp. 285–288).

Chopra, S., Hadsell, R., & LeCun, Y. (2005). Learning a similarity metric discriminatively, with application to face verification. In *2005 ieee computer society conference on computer vision and pattern recognition (cvpr'05)* (Vol. 1, pp. 539–546).

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Chung, J. S., Huh, J., Mun, S., Lee, M., Heo, H. S., Choe, S., ... Han, I. (2020). In defence of metric learning for speaker recognition. In *Interspeech.*

Chung, J. S., Nagrani, A., & Zisserman, A. (2018). Voxceleb2: Deep speaker recognition. *arXiv preprint arXiv:1806.05622*.

Davis, S. B., & Mermelstein, P. (1990). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. In *Readings in speech recognition* (pp. 65–74). Elsevier.

Dempsey, P. (2017). The teardown: Google home personal assistant. *Engineering & Technology*, *12*(3), 80–81.

Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4690–4699).

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. (2019). Neural spline flows. In *Advances in neural information processing systems* (pp. 7511–7522).

Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., ... others (2010). Building watson: An overview of the deepqa project. *AI magazine*, *31*(3), 59–79.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning.* MIT Press. (http://www.deeplearningbook.org)

Griffin, D., & Lim, J. (1984). Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *32*(2), 236–243.

Gruber, T. R. (2009). *Siri, a virtual personal assistant—bringing intelligence to the interface.* Jun.

Hernandez, F., Nguyen, V., Ghannay, S., Tomashenko, N., & Estève, Y. (2018). Ted-lium 3: twice as much data and corpus repartition for experiments on speaker adaptation. In *International conference on speech and computer* (pp. 198–208).

Hoogeboom, E., Berg, R. v. d., & Welling, M. (2019). Emerging convolutions for generative normalizing flows. *arXiv preprint arXiv:1901.11137*.

Ioffe, S. (2006). Probabilistic linear discriminant analysis. In *European conference on computer vision* (pp. 531–542).

Ito, K. (2017). *The lj speech dataset.* https://keithito.com/LJ-Speech-Dataset/. (Accessed: 2020-04-29)

Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., ... Kavukcuoglu, K. (2018). Efficient neural audio synthesis. *arXiv preprint arXiv:1802.08435*.

Kim, J., Kim, S., Kong, J., & Yoon, S. (2020). Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *arXiv preprint arXiv:2005.11129*.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., & Welling, M. (2016). Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems* (pp. 4743–4751).

Kumar, K., Kumar, R., de Boissiere, T., Gestin, L., Teoh, W. Z., Sotelo, J., ... Courville, A. C. (2019). Melgan: Generative adversarial networks for conditional waveform synthesis. In *Advances in neural information processing systems* (pp. 14910–14921).

Kyle, K. K. J. F. S., Jose, K. A. C. Y. B., & Sotelo, S. M. (2017). Char2wav: End-to-end speech synthesis. In *International conference on learning representations, workshop.*

Li, N., Liu, S., Liu, Y., Zhao, S., & Liu, M. (2019). Neural speech synthesis with transformer network. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, pp. 6706–6713).

Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., & Song, L. (2017). Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 212–220).

Panayotov, V., Chen, G., Povey, D., & Khudanpur, S. (2015). Librispeech: an asr corpus based on public domain audio books. In *Acoustics, speech and signal processing (ICASSP), 2015* *ieee international conference on* (pp. 5206–5210).

Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., … Miller, J. (2018). Deep voice 3: 2000-speaker neural text-to-speech. *Proc. ICLR*, 214–217.

Prenger, R., Valle, R., & Catanzaro, B. (2019). Waveglow: A flow-based generative network for speech synthesis. In *ICASSP 2019-2019 ieee international conference on acoustics, speech and signal processing (ICASSP)* (pp. 3617–3621).

Purington, A., Taft, J. G., Sannon, S., Bazarova, N. N., & Taylor, S. H. (2017). " Alexa is my new BFF" social roles, user satisfaction, and personification of the amazon echo. In *Proceedings of the 2017 chi conference extended abstracts on human factors in computing systems* (pp. 2853–2859).

Ramoji, S., Krishnan V, P., Singh, P., & Ganapathy, S. (2020). Pairwise discriminative neural plda for speaker verification. *arXiv preprint arXiv:2001.07034*.

Ribeiro, F., Florêncio, D., Zhang, C., & Seltzer, M. (2011). Crowdmos: An approach for crowdsourcing mean opinion score studies. In *Acoustics, speech and signal processing (ICASSP), 2011 ieee international conference on* (pp. 2416–2419).

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 815–823).

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., … others (2018). Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 ieee international conference on acoustics, speech and signal processing (ICASSP)* (pp. 4779–4783).

Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. In *Advances in neural information processing systems* (pp. 4077–4087).

Snyder, D., Garcia-Romero, D., Povey, D., & Khudanpur, S. (2017). Deep neural network embeddings for text-independent speaker verification. In *Interspeech* (pp. 999–1003).

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018). X-vectors: Robust dnn embeddings for speaker recognition. In *2018 ieee international conference on acoustics, speech and signal processing (ICASSP)* (pp. 5329–5333).

Sorensen, H. V., & Burrus, C. S. (1988). Efficient computation of the short-time fast fourier transform. In *ICASSP-88., international conference on acoustics, speech, and signal processing* (pp. 1894–1895).

Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015). Training very deep networks. In *Advances in neural information processing systems* (pp. 2377–2385).

Sztahó, D., Szaszák, G., & Beke, A. (2019). Deep learning methods in speaker recognition: a review. *arXiv preprint arXiv:1911.06615*.

Tachibana, H., Uenoyama, K., & Aihara, S. (2017). Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. *arXiv preprint arXiv:1710.08969*.

Tamamori, A., Hayashi, T., Kobayashi, K., Takeda, K., & Toda, T. (2017). Speaker-dependent wavenet vocoder. In *Proceedings of interspeech* (pp. 1118–1122).

Valle, R., Shih, K., Prenger, R., & Catanzaro, B. (2020). Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis. *arXiv preprint arXiv:2005.05957*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., … Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Veaux, C., Yamagishi, J., MacDonald, K., et al. (2016). Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*.

Wan, L., Wang, Q., Papir, A., & Moreno, I. L. (2018). Generalized end-to-end loss for speaker verification. In *2018 ieee international conference on acoustics, speech and signal processing (ICASSP)* (pp. 4879–4883).

Wang, F., Cheng, J., Liu, W., & Liu, H. (2018). Additive margin softmax for face verification. *IEEE Signal Processing Letters*, *25*(7), 926–930.

Wang, J., Wang, K.-C., Law, M. T., Rudzicz, F., & Brudno, M. (2019). Centroid-based deep metric learning for speaker recognition. In *ICASSP 2019-2019 ieee international conference on acoustics, speech and signal processing (ICASSP)* (pp. 3652–3656).

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., ... others (2017). Tacotron: A fully end-to-end text-to-speech synthesis model. *arXiv preprint arXiv:1703.10135*.

Ze, H., Senior, A., & Schuster, M. (2013). Statistical parametric speech synthesis using deep neural networks. In *2013 ieee international conference on acoustics, speech and signal processing* (pp. 7962–7966).

Zen, H., Dang, V., Clark, R., Zhang, Y., Weiss, R. J., Jia, Y., ... Wu, Y. (2019). Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*.

Zhu, X., Beauregard, G. T., & Wyse, L. L. (2007). Real-time signal estimation from modified short-time fourier transform magnitude spectra. *IEEE Transactions on Audio, Speech, and Language Processing*, *15*(5), 1645–1653.

CAPÍTULO 4

TTS-PORTUGUESE CORPUS: A CORPUS FOR SPEECH SYNTHESIS IN BRAZILIAN PORTUGUESE

Título:	**TTS-Portuguese Corpus: a corpus for speech synthesis in Brazilian Portuguese**
Autores:	**Edresson Casanova, Arnaldo Candido Junior, Christopher Shulby, Frederico Santos de Oliveira, João Paulo Teixeira, Moacir Antonelli Ponti e Sandra Aluísio**
Ano:	**2022**
Revista:	**Language Resources and Evaluation (LREV)**
Situação:	**Publicado**

Motivação:

Durante o desenvolvimento desse trabalho não existiam *datasets* publicamente disponíveis com quantidade de horas e qualidade suficiente para o treinamento de um modelo de síntese de fala no Português Brasileiro. Por esse motivo, nesse trabalho foi proposto e disponibilizado publicamente um *dataset* de síntese de fala composto por aproximadamente 10.5 horas de fala no Português Brasileiro.

Contribuições relevantes para a tese:

- Apresenta e disponibiliza publicamente o primeiro *dataset* para treinamento de modelos de síntese de fala baseados na abordagem de *deep learning* para o Português Brasileiro;

- Apresenta uma comparação entre dois modelos da literatura para a síntese de fala no Português Brasileiro, disponibilizando os *checkpoints* dos modelos;

- Os resultados dos experimentos são comparáveis aos obtidos no idioma Inglês e o atual estado da arte no idioma Português.

Lang Resources & Evaluation
https://doi.org/10.1007/s10579-021-09570-4

PROJECT NOTES

TTS-Portuguese Corpus: a corpus for speech synthesis in Brazilian Portuguese

Edresson Casanova[1] · Arnaldo Candido Junior[2] · Christopher Shulby[3] · Frederico Santos de Oliveira[4] · João Paulo Teixeira[5] · Moacir Antonelli Ponti[1] · Sandra Aluísio[1]

Accepted: 23 November 2021

Abstract Speech provides a natural way for human–computer interaction. In particular, speech synthesis systems are popular in different applications, such as personal assistants, GPS applications, screen readers and accessibility tools. However, not all languages are on the same level when in terms of resources and systems for speech synthesis. This work consists of creating publicly available resources for Brazilian Portuguese in the form of a novel dataset along with deep learning models

✉ Edresson Casanova
edresson@usp.br

Arnaldo Candido Junior
arnaldoc@utfpr.edu.br

Christopher Shulby
christopher@definedcrowd.com

Frederico Santos de Oliveira
fredericosantos@inf.ufg.br

João Paulo Teixeira
joaopt@ipb.pt

Moacir Antonelli Ponti
moacir@icmc.usp.br

Sandra Aluísio
sandra@icmc.usp.br

[1] Instituto de Ciências Matemáticas e de Computação, University of São Paulo, São Carlos, Brazil

[2] Federal University of Technology – Paraná (UTFPR), Medianeira, Brazil

[3] DefinedCrowd Corp., Seattle, USA

[4] Federal University of Mato Grosso, Cuiabá, Brazil

[5] Research Center in Digitalization and Intelligent Robotics (CEDRI) - Instituto Politecnico de Braganca, Bragança, Portugal

Published online: 12 January 2022

for end-to-end speech synthesis. Such dataset has 10.5 h from a single speaker, from which a Tacotron 2 model with the RTISI-LA vocoder presented the best performance, achieving a 4.03 MOS value. The obtained results are comparable to related works covering English language and the state-of-the-art in European Portuguese.

Keywords Corpora · Speech synthesis · TTS · Portuguese

1 Introduction

Speech synthesis systems have received a lot of attention in recent years due to the great advance provided by the use of deep learning, which allowed the popularization of virtual assistants, such as Amazon Alexa (Purington et al., 2017), Google Home (Dempsey, 2017) and Apple Siri (Gruber, 2009).

According to Tachibana et al. (2017), traditional speech synthesis systems are not easy to develop, because these are typically composed of many specific modules, such as, a text analyzer, a grapheme-to-phoneme converter, a duration estimator and an acoustic model. In summary, given an input text, the text analyzer module converts dates, currency symbols, abbreviations, acronyms and numbers into their standard formats to be pronounced or read by the system, i.e. carries out the text normalization and tackles problems like homographs, then with the normalized text the phonetic analyzer converts the grapheme into phonemes. In turn, the duration estimator estimates the duration of each phoneme. Finally, the acoustic model receives the phoneme representation sequence, the prosodic information about phoneme segments' length, the F0 contour and computes the speech signal (Teixeira et al., 2003; Ze et al., 2013). Several acoustic models have been proposed, such as the classical formant model (Klatt, 1980), Linear Prediction Coefficients (LPC) model, the Pitch Synchronous Overlap and Add (PSOLA) models (Charpentier & Stella, 1986) widely used in TTS engines like Microsoft Speech API. In addition, Hidden Markov Model (HMM) based synthesis is still a topic of research (Aroon & Dhonde, 2015; Braude et al., 2013; Tokuda et al., 2000), as well as a variety of Unit Selection Models (Siddhi et al., 2017; Wang & Georgila, 2011).

Deep learning (Goodfellow et al., 2016) allows to integrate all processing steps into a single model and connects them directly from the input text to the synthesized audio output, which is referred to as end-to-end learning. While neural models are sometimes criticized as difficult to interpret, several end-to-end trained speech synthesis systems (Kim et al., 2020; Sotelo et al., 2017; Ping et al., 2017; Shen et al., 2018; Tachibana et al., 2017; Valle et al., 2020; Wang et al., 2017) were shown to be able to estimate spectrograms from text inputs with promising performances. Due to the sequential characteristic of text and audio data, recurrent units were the standard building blocks for speech synthesis, such as in Tacotron 1 and 2 (Shen et al., 2018; Wang et al., 2017). In addition, convolutional layers showed good performance while reducing computational costs as implemented in DeepVoice3 and Deep Convolutional Text To Speech (DCTTS) methods (Ping et al., 2017; Tachibana et al., 2017).

Models based on deep learning require a greater amount of data for training, therefore, languages with low available resources are impaired. For this reason, most current TTS models are designed for the English language (Kim et al., 2020; Ping et al., 2017; Shen et al., 2018; Valle et al., 2020), which is a language with many open resources. In this work we propose to solve this problem for the Brazilian Portuguese language. Although there are some public datasets of speech synthesis for European Portuguese (Teixeira et al., 2001), due to the small amount of speech, approximately 100 min, makes the training of models based on deep learning unfeasible. In addition, simultaneously with this work, two datasets for automatic speech recognition for Portuguese, with good quality, were released. The CETUC dataset (Alencar & Alcaim, 2008), which was made publicly available by Quintanilha et al. (2020), has approximately 145 h of 100 speakers. In this dataset, each speaker uttered a thousand phonetically balanced sentences extracted from journalistic texts; on average each speaker spoke 1.45 h. The Multilingual LibriSpeech (MLS) dataset (Pratap et al., 2020) is derived from LibriVox audiobooks and consists of speech in eight languages including Portuguese. For Portuguese, the authors provided approximately 130 h of 54 speakers, an average of 2.40 h of speech per speaker. Although the quality of both datasets is good, both were made available with a sampling rate of 16 kHz and have no punctuation in their texts, making it difficult to apply them for speech synthesis. In addition, the amount of speech per speaker in both datasets is low, thus making it difficult to obtain a single-speaker dataset with a large vocabulary for single-speaker speech synthesis. For example, the LJ Speech dataset (Ito, 2017), which is derived from audiobooks and is one of the most popular open datasets for single-speaker speech synthesis in English, has approximately 24 h of speech.

In this article, we compare models of TTS available in the literature for a language with low available resources for speech synthesis. The experiments were carried out in Brazilian Portuguese and based on a single-speaker TTS. For this, we created a new public dataset, including 10.5 h of speech. Our contributions are twofold: (i) a new publicly available dataset with more than 10 h of speech recorded by a native speaker of Brazilian Portuguese; (ii) an experimental analysis comparing two publicly available TTS models in Brazilian Portuguese language. In addition, our results and discussions shed light on the matter of training end-to-end methods for a non-English language, in particular Portuguese, and the first public dataset and trained model for this language are made available.

This work is organized as follows. Section 2 presents related work on speech synthesis. Section 3 describes our novel audio dataset. Section 4 details the models and experiments performed. Section 5 compares and discusses the results. Finally, Section 6 presents conclusions of this work and future work.

2 Speech synthesis approaches

With the advent of deep learning, speech synthesis systems have evolved greatly, and are still being intensively studied. Models based on Recurrent Neural Networks, such as Tacotron (Wang et al., 2017), Tacotron 2 (Shen et al., 2018), Deep Voice 1

(Arik et al., 2017) and Deep Voice 2 (Arık et al., 2017), have gained prominence, but as these models use recurrent layers they have high computational costs. This has led to the development of fully convolutional models, such as DCTTS (Tachibana et al., 2017) and Deep Voice 3 (Ping et al., 2017), which sought to reduce computational cost while maintaining good synthesis quality.

Ping et al. (2017) proposed a fully convolutional model for speech synthesis and compared three different vocoders: Griffin–Lim (Griffin & Lim, 1984), WORLD Vocoder (Morise et al., 2016) and WaveNet (Van Den Oord et al., 2016). Their results indicated that WaveNet neural vocoder produced a more natural waveform synthesis. However, WORLD was recommended due to its better runtime, even though WaveNet had better quality. The authors further compared the proposed model (Deep Voice 3) with the Tacotron (Wang et al., 2017) and Deep Voice 2 (Arık et al., 2017) models.

Tachibana et al. (2017) proposed the DCTTS model, a fully convolutional model, consisting of two neural networks. The first, called Text2Mel (text to Mel spectrogram), which aims to generate a Mel spectrogram from an input text and the second, Spectrogram Super-resolution Network (SSRN), which converts a Mel spectrogram to the STFT (Short-time Fourier Transform) spectrogram (Benesty et al., 2011). DCTTS consists of only convolutional layers and uses dilated convolution (Kalchbrenner et al., 2016; Yu & Koltun, 2015) to take long, contextual information into account. DCTTS uses the vocoder RTISI-LA (Real-Time Iterative Spectrogram Inversion with Look-Ahead) (Zhu et al., 2007), which is an adaptation of the Griffin–Lim vocoder (Griffin & Lim, 1984), which aims to increase the speed of the synthesis by slightly sacrificing the quality of the audio generated.

Tacotron 1 (Wang et al., 2017) proposes the use of a single trained end-to-end Deep neural network. The model includes an encoder and a decoder. It uses an attention mechanism (Bahdanau et al., 2014) and also includes a post-processing module. This model uses convolutional filters, skip connections (Srivastava et al., 2015), and Gated Recurrent Units (GRUs) (Chung et al., 2014) neurons. Tacotron also uses Griffin–Lim (1984) algorithm to convert the STFT spectrogram to the wave form.

Tacotron 2 (Shen et al., 2018) combines Tacotron 1 with a modified WaveNet vocoder (Tamamori et al., 2017). Tacotron 2 is composed of a recurrent network of prediction resources from sequence to sequence that maps the incorporation of characters in Mel spectrograms, followed by a modified WaveNet model acting as a vocoder to synthesize waveforms in the time domain from those spectrograms. They also demonstrated that the use of Mel spectrograms as the conditioning input for WaveNet, instead of linguistic characteristics, allows a significant reduction in the size of the WaveNet architecture.

3 TTS-Portuguese Corpus

Portuguese is a language with few publicly available resources for speech synthesis. In Portuguese, as far as we know, there is no public dataset with a large amount of speech and quality available for speech synthesis. Although there are some public

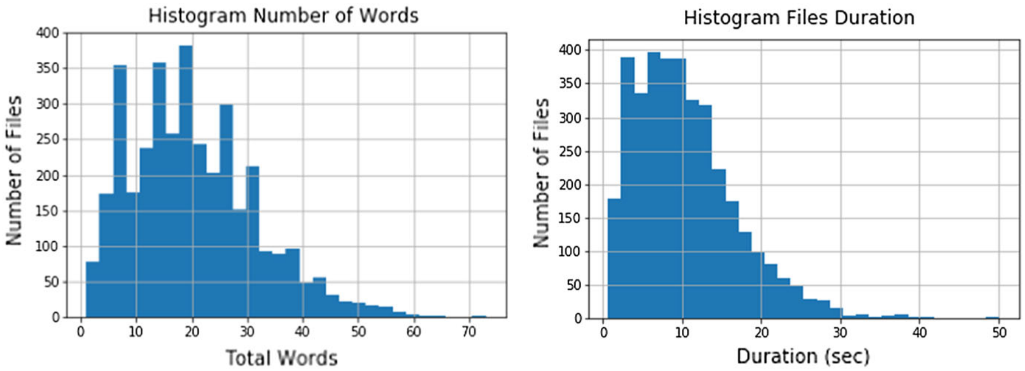


Fig. 1 The figure on the left shows the number of words and on the right the duration of each file

speech datasets for European Portuguese, for example (Teixeira et al., 2001), the work has a small amount of speech, approximately 100 min, which normally is not useful for training deep-learning models. On the other hand, Quintas and Trancoso (2020) explored the training of a model based on deep learning with an in-house dataset, called SSF1, which has approximately 14 h of speech in European Portuguese. Therefore, given the inexistence of an open dataset with a large amount of speech and quality for speech synthesis for Portuguese, we propose the TTS-Portuguese Corpus.

To create the TTS-Portuguese Corpus, public domain texts were used. Initially, seeking to reach a large vocabulary we extracted articles from the Highlights sections of Wikipedia for all knowledge areas. After this extraction, we separated the articles into sentences (considering textual punctuation) and randomly selected sentences from this corpus during the recording. In addition, we used 20 sets of phonetically balanced sentences, each set containing ten sentences proposed by Seara (1994). Finally, in order to increase the number of questions and introduce more expressive speech, we extracted sentences from Chatterbot-corpus[1], a corpus originally created for the construction of chatbots. Therefore, we decided both to have a large vocabulary and also to bring words from different areas. In addition, to have an expressive speech representation with the use of questions and answers from a chatbot dataset.

The recording was made by a male native Brazilian Portuguese speaker, not professional, in quiet environment but without acoustic isolation due to difficulties having access to studios. All the audios were recorded at a sampling frequency of 48 kHz and a 32-bit resolution.

In the dataset, each audio file has its respective textual transcription (phonetic transcription is not provided). The final dataset consists of a total of 71,358 words spoken by the speaker, 13,311 unique words, resulting in 3632 audio files and totaling 10 h and 28 min of speech. Audio files range in length from 0.67 to 50.08 s. The Fig. 1 shows two histograms regarding the number of words and the duration of each file.

[1] https://github.com/gunthercox/chatterbot-corpus/.

Table 1 Comparison between LJ Speech, SSF1 and TTS-Portuguese datasets in terms of language, duration, sampling rate and proportion of sentences: interrogative (Int), exclamatory (Exc) and declarative (Dec)

Dataset	Language	Duration (h)	Sampling rate (kHZ)	Int. (%)	Exc. (%)	Dec. (%)
LJ Speech	EN	24	22.05	0.58	0.35	99.07
SSF1	PT-PT	14	16	11	0.7	88.3
This work	PT-BR	10	48	3.42	0.38	96.96

To compare the TTS-Portuguese Corpus with datasets used in the literature for speech synthesis, we chose the LJ Speech dataset (Ito, 2017), which is one of the most widely used, publicly available datasets, for training single-speaker models in the English language. Additionally, we present the statistics for the SSF1 dataset, which is a corpus of European Portuguese explored in the work of Quintas and Trancoso (2020). Table 1 shows the language, duration, sampling rate and percentage of interrogative, exclamatory and declarative sentences in the LJ Speech, the SSF1 and the TTS-Portuguese datasets.

The TTS-Portuguese Corpus dataset has a smaller number of hours when compared to the others, it has 14 h less than the LJ Speech and 4 h less than the SSF1.

The sampling rate of 22.05 kHz is widely used in the training of TTS models based on deep learning. However, some works like Shen et al. (2018) use a sampling rate of 24 kHz. In addition, Kumar et al. (2020) showed that it is possible to obtain a 44.1 kHz TTS model by training the NU-GAN model on a dataset sampled at 44.1 kHz. Following this idea, we made available the TTS-Portuguese Corpus at 48 kHz, a sampling rate much higher than the datasets compared here. Finally, in the distribution of interrogative and exclamatory sentences, the TTS-Portuguese Corpus has less coverage for these sentences compared to the SSF1 in-house dataset. However, the TTS-Portugese Corpus has greater coverage of these sentences than the publicly available dataset LJ Speech.

The TTS-Portuguese Corpus[2] is open source, and publicly available under the terms of the license Creative Commons Attribution 4.0 (CC BY 4.0)[3].

4 Experiments

To evaluate the quality of the TTS-Portuguese Corpus in practice, we explored the speech synthesis using models of prominence in the literature. We chose the models: DCTTS (Tachibana et al., 2017) and Tacotron 2 (Shen et al., 2018).

Here, we compare the models DCTTS and Tacotron 2. To maintain results reproducible, we used open source implementations and tried to replicate related works as faithfully as possible. In the cases where hyper-parameters were not

[2] Official repository: https://github.com/Edresson/TTS-Portuguese-Corpus.

[3] https://creativecommons.org/licenses/by/4.0/.

specified, we empirically optimized those for our dataset. We have used the following implementations: DCTTS provided by Park (2018) and Tacotron 2 provided by Gölge (2019).

For all experiments, to speed up training, we initialized the model using the weights of the pre-trained model on English, using the LJ Speech dataset and we also use RTISI-LA (Zhu et al., 2007) as a vocoder, which is a variation of the Griffin–Lim (1984) vocoder.

Although the acquisition avoided external noise as best as possible, the audio files were not recorded in a studio setting. Therefore, some noise may be present in part of the files. To reduce the interference with our analysis, we applied RNNoise (Valin, 2017) in all audio files. RNNoise is based on Recurrent Neural Networks; more specifically Gated Recurrent Unit (Cho et al., 2014), and demonstrated good performance for noise suppression.

We report two experiments:

- Experiment 1: replicates the implementation of the DCTTS model, training the model for Brazilian Portuguese with the TTS-Portuguese Corpus. For this experiment, as reported in the DCTTS article, the model receives the text directly as input, so no phonetic transcription is used. As previously mentioned, the original DCTTS paper does not describe any normalization, so for the model to converge we tested different normalization options and decided to use, in all layers, 5% dropout and layer normalization (Ba et al., 2016). We did not use a fixed learning rate as described in the original article. Instead, we used a starting learning rate of 0.001 decaying using Noam's learning rate decay scheme (Vaswani et al., 2017).
- Experiment 2: this experiment explores Tacotron 2 (Shen et al., 2018) model, for that we use the Mozilla TTS implementation (Gölge, 2019). This model receives phonetic transcription as input instead of text directly. To perform phonetic transcription we use the Phonemizer[4] library that supports 121 languages and accents.

In experiment 1, two parts of the model are trained separately. The first part of the model, called Text2Mel, is responsible for generating a Mel spectrogram from the input text and this part of the model was induced using the composition of the functions: binary cross-entropy, L1 (Goodfellow et al., 2016) and guided attention loss (Tachibana et al., 2017). The second part, called SSRN, is responsible for the transformation of a mel spectrogram into the complete STFT spectrogram and applies super-resolution in the process and the loss function is composed of the functions L1 and binary cross-entropy.

In experiment 2, no guided attention is used, therefore, the loss function did not include the cost of attention. Since the network is trained end-to-end, the loss depends on the output of two network modules. The first module converts text into Mel spectrogram. The second module is a SSRN-like module called CBHG (1-D Convolution Bank Highway Network Bidirectional Gated Recurrent Unit).

[4] https://github.com/bootphon/phonemizer.

Table 2 Hardware specifications of the computers used in training

Specifications	Computer 1	Computer 2
Processor	i7-8700	i7-7700
RAM memory	16 GB	32 GB
Video Card	Nvidia GeForce Gtx Titan V	Nvidia GeForce Gtx 1080 TI
Operational system	Ubuntu 18.04	Windows 10

Table 3 Model training

Experiment	Training steps	Time
Experiment 1 (Text2Mel/SSRN)	2115 k/2019 k	4 days 19 h/5 days 22 h
Experiment 2	261 k	9 days 7 h

Table 2 shows the hardware specifications of the equipment used for model training. Experiment 1 was trained on computer 2, while experiment 2 were performed using computer 1.

Table 3 presents the training data from the experiments. The metrics presented in the table are: number of training steps, and the time required for training. It is important to note that experiment 1 is trained in two phases, both reported in the table: Text2Mel and SSRN.

5 Results and discussion

To compare and analyze our results we used the Mean Opinion Score (MOS) calculated following the work of Ribeiro et al. (2011). To calculate the MOS, the evaluators were asked to assess the naturalness of generated sentences on a five-point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). We chose 20 phonetically balanced sentences (Seara, 1994) not seen in the training, so that our analysis has a good phonetic coverage. However, in this set of sentences there are no interrogative and exclamatory sentences. These sentences were synthesized for each of our experiments. In addition, 20 samples with the pronunciation of these sentences by the original speaker were added as ground truth. Each sample was evaluated by 20 native evaluators. Our Models, synthesized audios, corpus and an interactive demo are public available.[5]

Table 4 presents the MOS values, with their respective 95% confidence intervals, for our experiments and for the best experiment of the Quintas and Trancoso (2020), which can also be seen in Fig. 2.

[5] https://edresson.github.io/TTS-Portuguese-Corpus/.

Table 4 MOS results

Experiment	MOS (rank)
Ground truth—SSF1	4.42 ± 0.60 (–)
Quintas and Trancoso (2020)	3.82 ± 0.69 (–)
Ground truth—Our	4.71 ± 0.16 (–)
Experiment 1	3.03 ± 0.34 (2)
Experiment 2	4.02 ± 0.27 (1)

The results of the main analysis indicate that experiment 2 (Mozilla TTS) presented the best MOS value (4.02). According to Ribeiro et al. (2011), the obtained value indicates a good quality audio, with a barely perceptible, but not annoying, distortion. On the other hand, experiment 1 presented a MOS of 3.03 indicating a perceptible and slightly annoying distortion in the audios.

With respect to previous results in the English language, Shen et al. (2018) (Tacotron 2) can be compared to our experiment 2. The authors trained their model on an in-house US English dataset (24.6 h), reaching a MOS of 4.52 ± 0.06. Considering the confidence intervals, our model reaches 4.29 in the best case and 3.75 in the worst case. Therefore, our model has a slightly lower MOS, and this can be justified as Shen et al. (2018) uses the WaveNet vocoder which achieves a higher quality in relation to the RTISI-LA/Griffin–Lim vocoder as shown in (Ping et al., 2017).

Finally, regarding DCTTS we obtained 3.03 ± 0.34 MOS. The original paper (Tachibana et al., 2017) had 2.71 ± 0.66 on the LJ Speech dataset. Therefore, the Tachibana et al. (2017) model can at best achieve a MOS of 3.37 and in the worst case 2.05. On the other hand, our model can at best achieve a MOS of 3.37 and in the worst case 2.69. Thus, the DCTTS model trained in the LJ Speech dataset and the TTS-Portuguese Corpus dataset showed similar MOS results.

It is also possible to compare our results with related works in European Portuguese. The current state of the art (SOTA) in European Portuguese (Quintas & Trancoso, 2020) achieved a MOS score of 3.82 ± 0.69 when training Tacotron 2 on the in-house SSF1 dataset. Considering the confidence intervals in the best case, the model of Quintas and Trancoso (2020) can achieve a MOS of 4.51 and in the worst case of 3.13. On the other hand, as previously discussed, our best model can reach 4.29 and 3.75 in the best and worst cases, respectively. These values are compatible since in the work of Quintas and Trancoso (2020) the authors used the neural vocoder WaveNet that generates speech with a higher quality. In addition, our confidence intervals are shorter, which may indicate that our evaluators agreed more during the evaluation. Furthermore, Quintas and Trancoso (2020) used only 8 evaluators in their MOS analysis, while in this work we used 20 evaluators; the number of evaluators can also have a impact on confidence intervals.

Comparing the Ground truth for the SSF1 dataset and the TTS-Portuguese Corpus, we can see that the TTS-Portuguese Corpus can vary from 4.87 to 4.55 in

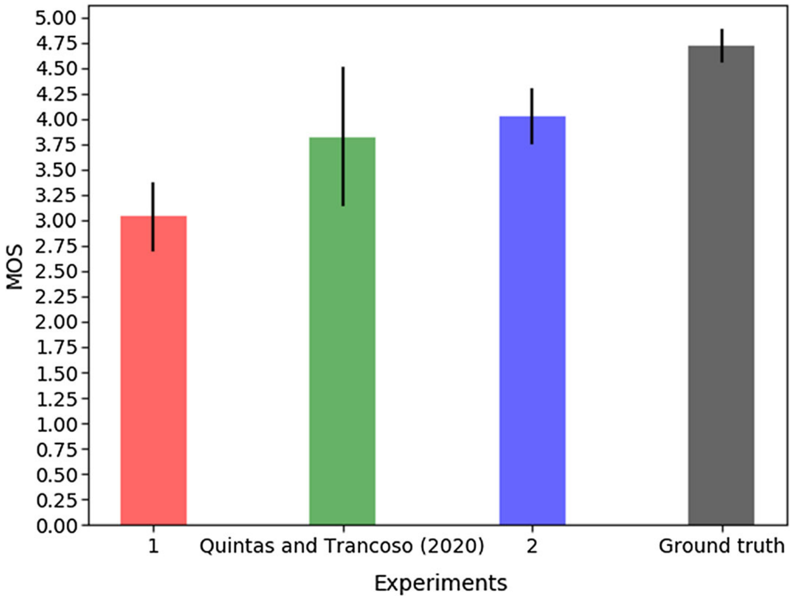


Fig. 2 MOS analysis chart

the best and worst cases, respectively. On the other hand, the MOS for the SSF1 dataset reported by Quintas and Trancoso (2020) ranges from 5.02 (a value above 5 can be justified by rounding) to 3.82. Considering this MOS analysis, the two datasets are comparable in terms of quality and naturalness.

6 Conclusions and future work

We found that it is possible to train a good quality speech synthesizer for Brazilian Portuguese using our dataset, reaching 4.02 MOS value. Our best results were based on Tacotron 2 model. We had MOS scores comparable to the SOTA paper that explores the use of deep learning in the European Portuguese language (Quintas & Trancoso, 2020), using a in-house dataset. In addition, our results are also comparable to works in the literature that used the English language.

To the best of our knowledge, this is the first publicly available single-speaker synthesis dataset for the language. Similarly, the trained models are a contribution to the Brazilian Portuguese language, since this language has limited open access models based on deep learning.

Future work can investigate the training of flow-based (Durkan et al., 2019; Hoogeboom et al., 2019; Kingma et al., 2016) TTS models, such as Flowtron (Valle et al., 2020), GlowTTS (Kim et al., 2020) and Flow-TTS (Miao et al., 2020), in the TTS-Portuguese Corpus dataset. Also, a more in-depth analysis of the datasets can be carried out to enrich it. For example, obtain the statistics of homographic-heterophone word pairs. Although time-consuming, it would provide a better learning context for the homograph disambiguation.

Acknowledgements This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior—Brasil (CAPES)—Finance Code 001, as well as CNPq (National Council of Technological and Scientific Development) Grants 304266/2020-5. This research was carried out at the Center for Artificial Intelligence (C4AI-USP), with support by the São Paulo Research Foundation (FAPESP Grant #2019/07665-4) and by the IBM Corporation. Also, we would like to thank the partial financial support for this paper provided by CEIA[6] (Artificial Intelligence Excellence Center, in English) via a funded project by the Cyberlabs Group[7]. We also gratefully acknowledge the support of NVIDIA corporation with the donation of the GPU used in part of the experiments presented in this research.

Authors' contributions EC, ACJ and SA proposed the methodology for collecting the dataset. EC, CS, ACJ, MAP and JPT proposed the experiments. EC and FSdO implemented and trained the models. All authors contributed to the final manuscript.

Funding This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior—Brasil (CAPES)—Finance Code 001, as well as CNPq (National Council of Technological and Scientific Development) Grant 304266/2020-5.

Data availability Dataset is available on the github of the project: https://github.com/Edresson/TTS-Portuguese-Corpus.

Code availability The trained models and demos are available on the github of the project: https://github.com/Edresson/TTS-Portuguese-Corpus.

Declarations

Conflict of interest The authors have no conflicts of interest to declare.

References

Alencar, V., & Alcaim, A. (2008). LSF and lPC-derived features for large vocabulary distributed continuous speech recognition in Brazilian Portuguese. In *2008 42ndAsilomar conference on signals, systems and computers* (pp. 1237–1241). IEEE.

Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Raiman, J., & Sengupta, S., & Ng, A. (2017). Deep voice: Real-time neural text-to-speech. *arXiv preprint*. http://arxiv.org/abs/170207825

Arık, S. O., Diamos, G., Gibiansky, A., Miller, J., Peng, K., Ping, W., Raiman, J., & Zhou, Y. (2017). Deep voice 2: Multi-speaker neural text-to-speech. *arXiv preprint*. http://arxiv.org/abs/170508947

Aroon, A., & Dhonde, S. (2015). Statistical parametric speech synthesis: A review. In *2015 IEEE 9th international conference on intelligent systems and control (ISCO)* (pp. 1–5). IEEE.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint*. http://arxiv.org/abs/160706450

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint*. http://arxiv.org/abs/14090473

Benesty, J., Chen, J., & Habets, E. A. (2011). *Speech enhancement in the STFT domain*. Springer Science & Business Media.

Braude, D. A., Shimodaira, H., & Youssef, A. B. (2013). Template-warping based speech driven head motion synthesis. In *Interspeech* (pp. 2763–2767).

[6] http://centrodeia.org/.

[7] https://cyberlabs.ai/.

Charpentier, F., & Stella, M. (1986). Diphone synthesis using an overlap-add technique for speech waveforms concatenation. In *ICASSP'86. IEEE international conferenceon acoustics, speech, and signal processing* (Vol. 11, pp. 2015–2018). IEEE.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *arXiv preprint*. http://arxiv.org/abs/14061078

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint*. http://arxiv.org/abs/14123555

Dempsey, P. (2017). The teardown: Google home personal assistant. *Engineering & Technology, 12*(3), 80–81.

Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. (2019). Neural spline flows. In *Advances in neural information processing systems* (pp. 7511–7522)

Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1). MIT Press.

Griffin, D., & Lim, J. (1984). Signal estimation from modified short-time Fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing, 32*(2), 236–243.

Gruber, T. R. (2009) Siri, a virtual personal assistant-bringing intelligence to the interface. In *Semantic technologies conference*.

Gölge, E. (2019). Deep learning for text to speech. https://github.com/mozilla/TTS

Hoogeboom, E., Van Den Berg, R., & Welling, M. (2019). Emerging convolutions for generative normalizing flows. *arXiv preprint*. http://arxiv.org/abs/190111137

Ito, K. (2017). The lj speech dataset. Retrieved April 29, 2020, from https://keithito.com/LJ-Speech-Dataset/

Kalchbrenner, N., Espeholt, L., Simonyan, K., van den Oord, A., Graves, A., & Kavukcuoglu, K. (2016). Neural machine translation in linear time. *arXiv preprint*. http://arxiv.org/abs/161010099

Kim, J., Kim, S., Kong, J., & Yoon, S. (2020). Glow-TTS: A generative flow for text-to-speech via monotonic alignment search. *arXiv preprint*. http://arxiv.org/abs/200511129

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., & Welling, M. (2016). Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems* (pp 4743–4751).

Klatt, D. H. (1980) Software for a cascade/parallel formant synthesizer. *The Journal of the Acoustical Society of America, 67*(3), 971–995.

Kumar, R., Kumar, K., Anand, V., Bengio, Y., & Courville, A. (2020) NU-GAN: High resolution neural upsampling with GAN. *arXiv preprint*. http://arxiv.org/abs/201011362

Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., Courville, A., Bengio, Y., & Sample R. N. N. (2017). Char2wav: End-to-end speech synthesis. In *International conference on learning representations, workshop*.

Miao, C., Liang, S., Chen, M., Ma, J., Wang, S., & Xiao, J. (2020). Flow-TTS: A non-autoregressive network for text to speech based on flow. In *ICASSP 2020–2020 IEEE international conference on acoustics, speech and signal processing(ICASSP)* (pp. 7209–7213). IEEE.

Morise, M., Yokomori, F., & Ozawa, K. (2016). World: A vocoder-based high-quality speech synthesis system for real-time applications. *IEICE Transactions on Information and Systems, 99*(7), 1877–1884.

Park, K. (2018). A tensorflow implementation of DC-TTS. https://github.com/Kyubyong/dc_tts

Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., Raiman, J., & Miller, J. (2017). Deep voice 3: 2000-speaker neural text-to-speech. *arXiv preprint*. http://arxiv.org/abs/171007654

Pratap, V., Xu, Q., Sriram, A., Synnaeve, G., & Collobert, R. (2020). Mls: A large-scale multilingual dataset for speech research. In *Proceedings of Interspeech 2020* (pp. 2757–2761).

Purington, A., Taft, J. G., Sannon, S., Bazarova, N. N., & Taylor, S. H (2017). "Alexa is my new BFF": Social roles, user satisfaction, and personification of the Amazon echo. In *Proceedings of the 2017 CHI conference extended abstracts on human factors in computing systems* (pp. 2853–2859).

Quintanilha, I. M., Netto, S. L., & Biscainho, L. W. P. (2020). An open-source end-to-end ASR system for Brazilian Portuguese using DNNs built from newly assembled corpora. *Journal of Communication and Information Systems, 35*(1), 230–242.

Quintas, S., & Trancoso, I. (2020). Evaluation of deep learning approaches to text-to-speech systems for European Portuguese. In *International conference on computational processing of the Portuguese language* (pp. 34–42). Springer.

Ribeiro, F., Florêncio, D., Zhang, C., & Seltzer, M. (2011). Crowdmos: An approach for crowdsourcing mean opinion score studies. In *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 2416–2419). IEEE.

Seara, I. (1994). Estudo estatístico dos fonemas do português brasileiro falado na capital de santa catarina para elaboração de frases foneticamente balanceadas. PhD thesis, Dissertação de Mestrado, Universidade Federal de Santa Catarina ...

Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., Chen, Z., Zhang, Y., Wang, Y., Skerrv-Ryan, R., & Saurous R. A. (2018). Natural TTS synthesis by conditioning WaveNet on Mel spectrogram predictions. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 4779–4783). IEEE.

Siddhi, D., Verghese, J. M., & Bhavik, D. (2017). Survey on various methods of text to speech synthesis. *International Journal of Computer Applications, 165*(6), 26–30.

Sotelo, J., Mehri, S., Kumar, K., Santos, J. F., Kastner, K., Courville, A., & Bengio, Y. (2017). Char2wav: End-to-end speech synthesis. In *International conference on learning representations, workshop*.

Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015). Training very deep networks. In *Advances in neural information processing systems* (pp. 2377–2385).

Tachibana, H., Uenoyama, K., & Aihara, S. (2017). Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. *arXiv preprint*. http://arxiv.org/abs/171008969

Tamamori, A., Hayashi, T., Kobayashi, K., Takeda, K., & Toda, T. (2017). Speaker-dependent WaveNet vocoder. In *Proceedings of Interspeech* (pp. 1118–1122).

Teixeira, J. P., Freitas, D., Braga, D., Barros, M. J., & Latsch, V. (2001). Phonetic events from the labeling the European Portuguese database for speech synthesis, FEUP/IPBDB. In *Seventh European conference on speech communication and technology*.

Teixeira, J. P., Freitas, D., & Fujisaki, H. (2003). Prediction of Fujisaki model's phrase commands. In *Eighth European conference on speech communication and technology*.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., & Kitamura, T. (2000). Speech parameter generation algorithms for HMM-based speech synthesis. In *2000 IEEE international conference on acoustics, speech, and signal processing. Proceedings (Cat. No. 00CH37100)* (Vol. 3, pp. 1315–1318). IEEE.

Valin, J. M. (2017). A hybrid DSP/deep learning approach to real-time full-band speech enhancement. *arXiv preprint*. http://arxiv.org/abs/170908243

Valle, R., Shih, K., Prenger, R., & Catanzaro, B. (2020). Flowtron: An autoregressive flow-based generative network for text-to-speech synthesis. *arXiv preprint*. http://arxiv.org/abs/200505957

Van Den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., & Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint*. http://arxiv.org/abs/160903499

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

Wang, W. Y., & Georgila, K. (2011). Automatic detection of unnatural word-level segments in unit-selection speech synthesis. In *2011 IEEE workshop on automatic speech recognition & understanding* (pp. 289–294). IEEE.

Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., & Bengio, S., & Le, Q. V. (2017). Tacotron: A fully end-to-end text-to-speech synthesis model. *arXiv preprint*. http://arxiv.org/abs/170310135

Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint*. http://arxiv.org/abs/151107122

Ze, H., Senior, A., & Schuster, M. (2013). Statistical parametric speech synthesis using deep neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing* (pp. 7962–7966). IEEE.

Zhu, X., Beauregard, G. T., & Wyse, L. L. (2007). Real-time signal estimation from modified short-time fourier transform magnitude spectra. *IEEE Transactions on Audio, Speech, and Language Processing, 15*(5), 1645–1653.

CAPÍTULO

5

SC-GLOWTTS: AN EFFICIENT ZERO-SHOT MULTI-SPEAKER TEXT-TO-SPEECH MODEL

Título:	**SC-GlowTTS: an Efficient Zero-Shot Multi-Speaker Text-To-Speech Model**
Autores:	**Edresson Casanova, Christopher Shulby, Eren Gölge, Nicolas Michael Müller, Frederico Santos de Oliveira, Arnaldo Candido Junior, Anderson da Silva Soares, Sandra Maria Aluisio, Moacir Antonelli Ponti**
Ano:	**2021**
Conferência:	**INTERSPEECH 2021**
Situação:	**Publicado**

Motivação:

Apesar dos avanços recentes, a síntese de fala *zero-shot multi-speaker* ainda é um problema em aberto, em particular no que diz respeito à diferença entre similaridade da fala gerada para locutores vistos e não vistos no treinamento dos modelos. Além disso, recentemente os *normalizing flows* (ou modelos *flow-based*) foram aplicados com sucesso na área de síntese de fala, alcançando resultados no estado da arte (VALLE *et al.*, 2020; KIM *et al.*, 2020). Apesar disso, os modelos de síntese de fala *zero-shot multi-speaker* ainda eram fortemente baseados no modelo Tacotron 2. Por fim, os modelos de síntese de fala *zero-shot multi-speaker* ainda exigiam uma grande quantidade de locutores para o treinamento, inviabilizando a obtenção de modelos de boa qualidade em idiomas com poucos recursos. Por esses motivos, nesse trabalho explorou-se a aplicação de *normalizing flows* em síntese de fala *zero-shot multi-speaker* bem como o treinamento do modelo com uma quantidade menor de locutores.

Contribuições relevantes para a tese:

- Apresenta uma nova arquitetura para síntese de fala *zero-shot multi-speake*r, apresentando resultados no estado da arte e permitindo a síntese de alta qualidade, em tempo real;
- Mostra que o ajuste fino de um *vocoder* baseado em GAN utilizando os espectrogramas preditos pelo modelo de síntese de fala para o conjunto de treinamento aumenta significativamente a qualidade e similaridade para novos locutores;
- Mostra que a abordagem proposta alcança resultados comparáveis ao estado da arte utilizando apenas 11 locutores durante o treinamento do modelo.

SC-GlowTTS: an Efficient Zero-Shot Multi-Speaker Text-To-Speech Model

Edresson Casanova[1]*, Christopher Shulby*[2]*, Eren Gölge*[3]*, Nicolas Michael Müller*[4]*, Frederico Santos de Oliveira*[5]*, Arnaldo Candido Junior*[6]*, Anderson da Silva Soares*[5]*, Sandra Maria Aluisio*[1]*, Moacir Antonelli Ponti*[1]

[1] Instituto de Ciências Matemáticas e de Computação, University of São Paulo, São Carlos/SP, Brazil
[2] DefinedCrowd Corp., Seattle, WA, USA
[3] Coqui, Berlin, Germany
[4] Fraunhofer AISEC, Garching near Munich, Germany
[5] Federal University of Goias, Goiânia, GO, Brazil
[6] Federal University of Technology – Paraná, Medianeira, PR, Brazil

edresson@usp.br

Abstract

In this paper, we propose SC-GlowTTS: an efficient zero-shot multi-speaker text-to-speech model that improves similarity for speakers unseen during training. We propose a speaker-conditional architecture that explores a flow-based decoder that works in a zero-shot scenario. As text encoders, we explore a dilated residual convolutional-based encoder, gated convolutional-based encoder, and transformer-based encoder. Additionally, we have shown that adjusting a GAN-based vocoder for the spectrograms predicted by the TTS model on the training dataset can significantly improve the similarity and speech quality for new speakers. Our model converges using only 11 speakers, reaching state-of-the-art results for similarity with new speakers, as well as high speech quality.

Index Terms: zero-shot multi-speaker TTS, text-to-speech, multi-speaker modeling, zero-shot voice conversion.

1. Introduction

Text-to-Speech (TTS) systems have received a lot of attention in recent years due to the great advances by deep learning, which have allowed for the popularization of voice applications such as virtual assistants. Most TTS systems were tailored from a single speaker voice, but there is current interest in synthesizing voices for new speakers, not seen during training, employing only a few seconds of speech samples. This approach is called zero-shot multi-speaker TTS (ZS-TTS) as in [1, 2, 3, 4].

ZS-TTS was first proposed [1] by extending the DeepVoice 3 [5]. Also, [2] explored Tacotron 2 [6] using external embeddings extracted from a trained speaker encoder using a generalized end-to-end loss (GE2E) [7], resulting in a model that can generate speech, resembling the target speaker. Similarly, [3] explored Tacotron 2 with different speaker embeddings methods. The authors showed that LDE [8] embeddings improved the similarity and synthesized a more natural speech for novel speakers when compared to X-vector [9] embeddings. The authors in [3] also showed that training a gender-dependent model improves the similarity for unseen speakers.

In this context, a major issue is the similarity gap between observed and unobserved speakers during training. In an attempt to reduce this gap, Attentron [4] proposed a fine-grained encoder with an attention mechanism for extracting detailed styles from various reference samples and a coarse-grained encoder. As a result of using several reference samples instead of one, they achieved a better similarity for unseen speakers.

Despite the recent results, zero-shot multi-speaker TTS remains an open problem in particular concerning the difference in the quality of seen and unseen speakers. Also, current approaches rely heavily on Tacotron 2, while there is potential to improve results with the use of flow-based methods [10]. In this context, FlowTron [11] allowed for the manipulation of multiple aspects of speech, such as pitch, tone, speech rate, cadence, and accent. Also, [12] proposed GlowTTS reaching similar quality to Tacotron 2 but with an increase in speed of 15.7 times while permitting speech velocity manipulation.

In this paper, we propose a novel method, Speaker Conditional GlowTTS (SC-GlowTTS), for zero-shot learning of unseen speakers. Our model relies on GlowTTS [12] for the part that converts input characters to spectrograms. SC-GlowTTS uses an external speaker encoder based on Angular Prototypical loss [13], to learn speaker embedding vectors, and adapts the HiFi-GAN [14] vocoder to convert the output spectrograms to the waveform. Our contribution is as follows:

- A novel zero-shot multi-speaker TTS approach that achieves state-of-the-art results with just 11 speakers in the training set;
- An architecture that enables high quality and faster than real-time speech synthesis in the zero-shot multi-speaker TTS setting;
- Adjusting a GAN-based vocoder for the spectrograms predicted by the TTS model on the training dataset, in order to significantly improve the similarity and speech quality for new speakers.

The audio samples for each of our experiments are available on the demo web-site[1]. In addition, for reproducibility the implementation is available at the Coqui TTS[2], and checkpoints of all experiments are available at the Github repository[3].

2. Speaker Conditional GlowTTS Model

Speaker Conditional Glow-TTS (SC-GlowTTS) builds upon GlowTTS, but includes several novel modifications. In addition to the GlowTTS's transformer-based encoder network, we explore a residual dilated convolutional network [15] and gated convolutional network [16]; to our knowledge, used for the first time in this context. Our convolutional residual encoder is based on [15], however we used the Mish [17] instead of ReLU activation function. On the other hand, our gated convolutional

[1] https://edresson.github.io/SC-GlowTTS/

[2] https://github.com/coqui-ai/TTS

[3] https://github.com/Edresson/SC-GlowTTS

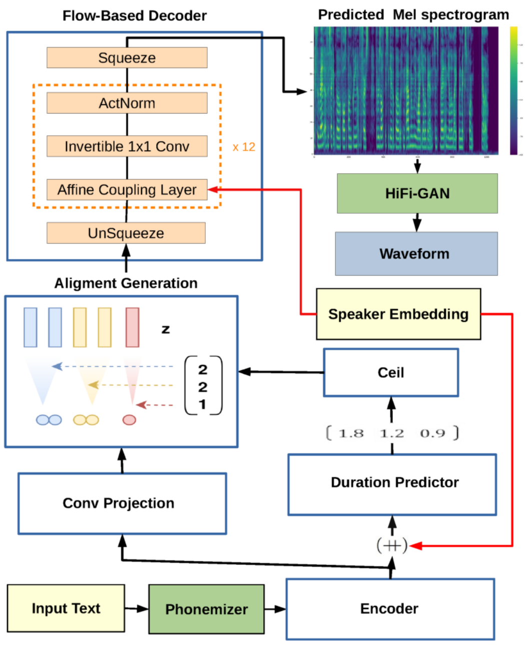


Figure 1: *Speaker Conditional GlowTTS General Architecture.*

network [16] consists of 9 convolutional blocks and each block includes a dropout layer, a 1D convolution, and a layer normalization [18]. We use kernel size 5, dilation rate 1, and 192 channels in all convolutional layers. A flow-based decoder is used with the same architecture and configuration as the GlowTTS model. However, to transform it into a zero-shot TTS model, we include speaker embeddings in the affine coupling layers on all 12 decoder blocks. We also used the FastSpeech's duration predictor network [19] to predict character durations. To capture different speech characteristics of different speakers, we added speaker embeddings to the input of the duration predictor. Finally, the HiFi-GAN [14] is used as a vocoder.

The SC-GlowTTS model, during inference, is illustrated in Figure 1, where $(+\!\!+)$ indicates concatenation. During training, the model uses the Monotonic Alignment Search (MAS) [12], where the decoder's objective is to condition the mel spectrogram and an input speaker embedding in a P_Z prior distribution. The purpose of MAS is to align the P_Z prior distribution with the encoder's output. During inference, MAS is not used, instead, the P_Z prior distribution and alignment are predicted by the text encoder and the duration predictor network. Finally, a latent variable Z is sampled from the prior distribution P_Z. Using the inverted decoder and the speaker embeddings, a mel spectrogram is synthesized in parallel, transforming the latent variable Z via the flow-based decoder.

For brevity, we denominate the SC-GlowTTS model with the transformer, residual convolution, and gated convolution based encoders as SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated model, respectively.

3. Experiments

3.1. Speaker Encoder

Our speaker encoder is a stack of 3 LSTM layers with a linear output layer, similar to [7]. We use 768 LSTM units and 256 units for the linear layer. For training, we used audios sampled at 16 kHz and extracted mel spectograms using a 1024ms window using the Fast Fourier Transform (FFT), with a hop length of 256 and 1024 FFT components, from which we retain only 80 mel coefficients. Optimization was carried out using the Angular Prototypical [13] loss function different than the original work. The optimizer RAdam [20] was used during 320k steps using 64 speakers per batch, with 10 samples of each speaker and a learning rate of 10^{-4}.

3.2. Zero-Shot Multi-Speaker Tacotron 2

We compare our approach with Tacotron 2. Following the proposal of [2], [21] and [3] we use local sensitive attention [6]. We concatenate the speaker embeddings to the input of the attention module as in [2, 3], given that the latter showed this was adequate for a gender-independent Tacotron model. To alleviate possible issues in the attention module we use Double Decoder Consistency (DDC) [22] with gradual training [23] and guided attention [24]. In Tacotron, the number of output frames per decoder iteration is called the reduction rate (R) [23, 6]. The idea of the DDC is to combine two decoders with different reduction factors. One decoder (coarse) works with a higher R and another decoder (fine) works with a smaller R value. Gradual training simply starts training with a larger R and decreases it during the training. In our experiments, we use $R = 7$ for the coarse decoder, while for the fine decoder we used gradual training, starting from $R = 7$ and decreasing it as follows: $R = 5$ at step 10k; $R = 3$ at step 25k; $R = 2$ at step 70k.

3.3. Audio datasets

Our speaker encoder was trained with all partitions of the LibriSpeech dataset [25], the English version of Common Voice [26], the VCTK and VoxCeleb (v1 and v2) datasets [27], totaling approximately 25k speakers.

Our zero-shot multi-speaker TTS model is trained using VCTK [28] dataset, an English language dataset containing 44 hours of speech and 109 speakers, sampled at 48KHz. Each speaker pronounces approximately 400 sentences. Preprocessing was carried out in order to remove long periods of silence. We applied voice activity detection (VAD) using Webrtcvad toolkit[4]. We have divided the VCTK dataset into: train, validation (containing the same speakers as the train set) and test. For the test set, we selected 11 speakers not present in the validation or training set; following the proposal by [2], we selected 1 representative from each accent totaling 7F/4M (speakers 225, 234, 238, 245, 248, 261, 294, 302, 326, 335 and 347). For the HiFi-GAN [14] vocoder initial training uses *train-clean-100* and *train-clean-360* partitions of the LibriTTS [29] dataset.

3.4. Experimental setup

We carried out four training experiments:

- **Experiment 1:** Tacotron zero-shot model, described in Section 3.2, trained for 210k steps.
- **Experiment 2:** SC-GlowTTS-Trans model trained for 150k steps.
- **Experiment 3:** SC-GlowTTS-Res model trained for 150k steps.
- **Experiment 4:** SC-GlowTTS-Gated model trained for 150k steps.

[4] https://github.com/wiseman/py-webrtcvad

In all experiments, we used RAdam [20] with batch size 128, an initial learning rate of 10^{-3}, and Noam's learning rate schedule [30] with 4000 warmup steps. We use the same configuration to extract the mel spectrograms from the speaker encoder, detailed in the Section 3.1 but with 22khz sampling rate. We use the VCTK dataset in all our experiments using the training, validation, and test partitions as specified in Section 3.3 and we use the validation set to choose the best checkpoint for each experiment comparing the loss value.

In all experiments, we choose to use phonemes as input instead of text. Specifically, we used the Phonemizer tool[5], which supports several languages. In addition, we add a blank token between each of the phonemes in the input sentence for the GlowTTS-based models, as suggested by the original work[6] [12].

HiFi-GAN v2 model was used as a vocoder, due to its effective speed/quality trade-off. As a starting point, we used the model provided by the authors trained for 500k steps with the LJ Speech [31] dataset. We first trained the HiFi-GAN model for 75k steps with the LibriTTS dataset. Afterward, the model is adjusted for other 190k steps using the VCTK dataset, using the training and validation partitions as specified in Section 3.3.

[14] showed that adjusting the HiFi-GAN model with the spectrogram of the TTS model, improves quality for a single speaker. However, it remains an open question whether it improves; (i) the quality in multi-speaker, (ii) speech similarity for unseen speakers in ZS-TTS settings. To answer this, our TTS models synthesize each of the sentences in the training and validation splits of VCTK dataset. We enabled teacher forcing to keep the alignments between predicted spectrogram frames and the input phonemes. For SC-GlowTTS we use the MAS to align the decoder output with the encoder output. Using these spectrograms extracted from each model, we fine-tuned the checkpoint initially trained with the LibriTTS dataset, for an additional 190k steps, producing the fine-tuned HiFi-GAN (HiFi-GAN-FT).

4. Results and Discussion

In this paper, the synthesized speech quality is evaluated using mean opinion score (MOS) study, following [32]. MOS scores were obtained with rigorous crowdsourcing [33]. For the MOS calculation, 15 professional collaborators per audio were invited from a total of 68 unique contributors (35F/33M). To compare the similarity between the synthesized voice and the original speaker, we calculate the Speaker Encoder Cosine Similarity (SECS). The SECS consists of calculating the cosine similarity between the embeddings of two audios extracted from the speaker encoder. It ranges from -1 to 1, and a larger value indicates a stronger similarity [3]. Following [4], we compute SECS using the speaker encoder of the Resemblyzer [7, 34] package; thus, allowing comparison with those studies. We also report the MOS similarity (Sim-MOS) following the work of [2] and [4].

We also compare the run-time of each model by calculating the Real Time Factor (RTF) on a CPU and GPU. For speed tests we used a machine with an NVIDIA GeForce GTX Titan V GPU, an Intel (R) Xeon (R) CPU E5-2603 v4 @ 1.70GHz processor with 6 CPU cores and 15 Gb of RAM. The training was carried out on an NVIDIA V100 GPU. Also, RTF was calculated considering the full synthesis run, from the input phonemes to the output waveform. We synthesize 15 different sentences as in [35] 10 times for each of the 11 speakers of the VCTK test set and calculated the average.

As a reference sample for the extraction of speaker embeddings, we use the fifth sentence of the VCTK (i.e, speakerID_005.txt), since all test speakers uttered it and because it is a long sentence (20 words). In this way, all speakers are presented in the zero-shot multi-speaker TTS model by a reference sample with the same number of words and speech content.

For the calculation of MOS and SECS we randomly drew 55 sentences from the *test-clean* subset of the LibriTTS, considering only sentences with more than 20 words. We randomly select five sentences for each of the 11 test speakers, ensuring that all 55 test sentences are synthesized and that all the test speakers are considered. As ground truth, we select 5 audios randomly for each of the 11 test speakers (55 in total), only audios with more than 20 words are studied.

On the other hand, for the SECS ground truth, we compared the 55 audios chosen at random (explained above) with the reference audios used to synthesize the sentences (fifth sentence of the VCTK dataset for each of the test speakers).

Table 1 shows the RTF in CPU and GPU, MOS with 95% confidence intervals and SECS for all of our experiments. Speed tests show that the fastest model on both CPU and GPU is SC-GlowTTS-Gated, followed by the SC-GlowTTS-Res model. The SC-GlowTTS-Trans model is the slowest of the SC-GlowTTS family, however, still much faster than Tacotron 2. Despite this, with the integration with the HiFi-GAN vocoder all models are real time in both CPU and GPU.

SECS score of the ground truth reached 0.9222 because it compares the sample used as a reference with other real speech samples of the same speaker. This value is intended to show an upper bound for SECS, i.e., a model that perfectly "copies" the voice of a target speaker.

The best SECS for synthesis with the HiFi-GAN vocoder (without fine-tuning) was obtained by the SC-GlowTTS-Trans model (experiment 2), followed by Tacotron 2 (experiment 1). The SC-GlowTTS-Res model (experiment 3) achieved the third-best SECS being only better than SC-GlowTTS-Gated (experiment 2). Using the HiFi-GAN-FT, the SC-GlowTTS-Trans model also obtained the best SECS, followed by the SC-GlowTTS-Res model. The SC-GlowTTS-Gated model reached the third-best SECS being only better than the Tacotron 2 model. We found that the fine-tuning of the HiFi-GAN vocoder in the spectrograms extracted from the TTS models significantly improves SECS for the new speakers. The SECS increased from 0.7589 to 0.7791, 0.7641 to 0.8046, 0.7440 to 0.7969 and 0.7432 to 0.7849, respectively, for the models Tacotron 2, SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated.

For Sim-MOS the results are similar to those of SECS. However, there are some differences which can be explained by the overlapping of the Sim-MOS confidence intervals between the experiments. Improvement with the use of HiFi-GAN-FT can also be seen in all experiments.

Finally, we compare our results with those presented by the Attentron model. In [4], the authors reported SECS values, also calculated by the speaker encoder that we use. Although the authors use only 8 speakers (4F/4M) for the test and we use 11 speakers, we believe the comparison is fair and leaves no selection criteria undefined. The Attentron model in zero-shot mode reached a SECS of only 0.731. Such a model uses multiple samples to synthesize speech instead of just one, performing few-shot TTS with 8 reference samples. This approach achieves a SECS of 0.788, slightly lower than our best SECS,

[5] https://github.com/bootphon/phonemizer/

[6] https://github.com/jaywalnut310/glow-tts

Table 1: *Real Time Factor, MOS and Sim-MOS with 95% confidence intervals and the SECS for all our experiments.*

Experiment - Model	**Vocoder**	**RTF (CPU - GPU)**	**SECS**	**MOS**	**Sim-MOS**
Ground Truth	–	–	0.9236	4.12 ± 0.06	4.127 ± 0.06
Attentron ZS [4]	WaveRNN	–	(0.731)	(3.86 ± 0.05)	(3.30 ± 0.06)
1 - Tacotron 2	HiFi-GAN	0.5782 - 0.2485	0.7589	3.57 ± 0.08	3.867 ± 0.08
	HiFi-GAN-FT	-	0.7791	3.74 ± 0.08	3.951 ± 0.07
2 - SC-GlowTTS-Trans	HiFi-GAN	0.3612 - 0.1557	0.7641	3.65 ± 0.07	3.905 ± 0.07
	HiFi-GAN-FT	-	**0.8046**	3.78 ± 0.07	**3.999 ± 0.07**
3 - SC-GlowTTS-Res	HiFi-GAN	0.3597 - 0.1545	0.7440	3.45 ± 0.09	3.828 ± 0.08
	HiFi-GAN-FT	-	0.7969	3.70 ± 0.07	3.916 ± 0.07
4 - SC-GlowTTS-Gated	HiFi-GAN	0.3474 - 0.1437	0.7432	3.55 ± 0.08	3.852 ± 0.08
	HiFi-GAN-FT	-	0.7849	**3.82 ± 0.07**	3.952 ± 0.07

0.8046. Despite the advantage of the few-shot approach, our model still achieves a higher SECS than Attentron. The authors also reported the Sim-MOS reaching 4.83 ± 0.02 for ground truth speech, zero-shot mode Attentron reaches 3.30 ± 0.06 and few-shot mode 3.57 ±0.05. Considering these values, our best model was superior at 3,999 ± 0.07. Furthermore, the results of our model are closer to the ground truth being only 0.128 smaller while in [4] the best experiment's difference is 1.26.

For the MOS, ground truth speech reached 4.12. The SC-GlowTTS-Gated model with the HiFi-GAN-FT vocoder was the closest, reaching a MOS of 3.82. Moreover, as in SECS, where the HiFi-GAN-FT vocoder improved speech similarity, the best MOS was achieved using the same vocoder. With the adjustment of the HiFi-GAN vocoder in the spectrograms extracted from the TTS model, the MOS for new speakers increased significantly from 3.57 to 3.74, 3.65 to 3.78, 3.45 to 3.70, 3.55 to 3.82, respectively, for all models Tacotron 2, SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated. Our MOS values are on par with the other state-of-the-art ZS-TTS models such as [3, 4].

5. SC-GlowTTS performance with few speakers

To emulate a scenario with few speakers, we reflect our test set by selecting a subset of the VCTK dataset training set. This new training set consists of 11 speakers, 7F/4M. We selected 1 representative for each accent, except for the "New Zealand" accent that has only one speaker and it is in our test set, so we added an "American" speaker instead, the chosen speakers were 229, 249, 293, 313, 301, 374, 304, 316, 251, 297 and 323. From this new training set, we have selected random samples to use as a validation set. As a test set, we use the same one defined in Section 3.3.

We use the SC-GlowTTS-Trans model and train it with the LJSpeech [31] dataset for 290k steps. This pre-training in a single-speaker dataset was carried out to prime the encoder of the model in a larger vocabulary. We fine-tuned the SC-GlowTTS-Trans model in the new training set with only 11 speakers for 70k steps and using the validation set, we selected the best checkpoint as step 66k. In addition, using the HiFi-GAN model trained in the LibriTTS dataset for 75k steps, we adjusted for other 95k steps using the same technique. This new experiment resulted in a SECS of 0.7707, a MOS of 3.71 ± 0.07 and Sim-MOS of 3.93 ± 0.08. These results are compatible with SECS of 0.7791, MOS 3.74 and Sim-MOS 3.951 ± 0.07 achieved by Tacotron 2, which used a much larger set of 98 speakers. Therefore, our SC-GlowTTS-Trans model converges with a 9.8 times smaller dataset, with comparable performance to Tacotron 2. We believe that this is an important step forward especially for ZS-TTS in low-resource languages.

6. Zero-Shot Voice Conversion

As in the original GlowTTS [12] model, we do not provide any information about the speaker's identity to the model encoder, so the distribution predicted by the encoder is forced to be independent of the speaker identities. Therefore, like GlowTTS, SC-GlowTTS can convert voices using only the model's decoder. However, in our work, we condition SC-GlowTTS with external speaker embeddings. It enables our model to resemble the voice for speakers not seen in the training by performing a zero-shot voice conversion. Samples of the zero-shot voice conversion are present on the demo page[7].

7. Conclusions and future work

In this work, we present a novel method, SC-GlowTTS, achieving state-of-the-art ZS-TTS results. We explored three different encoders for the SC-GlowTTS model and showed that a transformer-based encoder gave the best similarity for speakers not seen in the training. Our SC-GlowTTS models are superior to Tacotron 2. Also, when combined with an external speaker encoder, SC-GlowTTS models can perform ZS-TTS with only 11 speakers in the training set. Finally, we found that the adjustment of the HiFi-GAN vocoder in the spectrograms predicted by the TTS model in the training and validation set can significantly improve the similarity and the quality of the synthesized speech (MOS) for speakers not seen in the training. As future work, following the work of [4], we intend to extend the SC-GlowTTS as a few-shot approach.

8. Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Also, we would like to thank the CyberLabs[8] for financial support for this paper and Defined-Crowd for making industrial-level MOS testing so easily available. Finally, we would like to thank all contributors to the Coqui TTS repository[9], this work was only possible thanks to the commitment of all.

[7] https://edresson.github.io/SC-GlowTTS/

[8] https://cyberlabs.ai/

[9] https://github.com/coqui-ai/TTS

9. References

[1] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 019–10 029.

[2] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.

[3] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6184–6188.

[4] S. Choi, S. Han, D. Kim, and S. Ha, "Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding," *arXiv preprint arXiv:2005.08484*, 2020.

[5] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," *arXiv preprint arXiv:1710.07654*, 2017.

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

[7] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[8] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," *arXiv preprint arXiv:1804.05160*, 2018.

[9] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[10] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in neural information processing systems*, 2016, pp. 4743–4751.

[11] R. Valle, K. Shih, R. Prenger, and B. Catanzaro, "Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis," *arXiv preprint arXiv:2005.05957*, 2020.

[12] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *arXiv preprint arXiv:2005.11129*, 2020.

[13] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Interspeech*, 2020.

[14] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *arXiv preprint arXiv:2010.05646*, 2020.

[15] J. Vainer and O. Dušek, "Speedyspeech: Efficient neural speech synthesis," *arXiv preprint arXiv:2008.03802*, 2020.

[16] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *International conference on machine learning*. PMLR, 2017, pp. 933–941.

[17] D. Misra, "Mish: A self regularized non-monotonic neural activation function," *arXiv preprint arXiv:1908.08681*, 2019.

[18] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[19] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Advances in Neural Information Processing Systems*, 2019, pp. 3171–3180.

[20] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," *arXiv preprint arXiv:1908.03265*, 2019.

[21] D. Paul, Y. Pantazis, and Y. Stylianou, "Speaker conditional wavernn: Towards universal neural vocoder for unseen speaker and recording conditions," *arXiv preprint arXiv:2008.05289*, 2020.

[22] E. Gölge. (2020) Solving attention problems of tts models with double decoder consistency. [Online]. Available: https://erogol.com/solving-attention-problems-of-tts-models-with-double-decoder-consistency/

[23] ——. (2019) Gradual training with tacotron for faster convergence. [Online]. Available: https://erogol.com/gradual-training-with-tacotron-for-faster-convergence/

[24] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," *arXiv preprint arXiv:1710.08969*, 2017.

[25] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[26] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.

[27] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech 2018*, 2018, pp. 1086–1090. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1929

[28] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2016.

[29] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[31] K. Ito *et al.*, "The lj speech dataset," 2017.

[32] F. Ribeiro, D. Florêncio, C. Zhang, and M. Seltzer, "Crowdmos: An approach for crowdsourcing mean opinion score studies," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2416–2419.

[33] Mos testing. DefinedCrowd Corp. [Online]. Available: https://www.definedcrowd.com/evaluation-of-experience/

[34] C. Jemine *et al.*, "Master thesis: Real-time voice cloning," 2019.

[35] K. Peng, W. Ping, Z. Song, and K. Zhao, "Non-autoregressive neural text-to-speech," in *International Conference on Machine Learning*. PMLR, 2020, pp. 7586–7598.

CAPÍTULO

6

YOURTTS: TOWARDS ZERO-SHOT MULTI-SPEAKER TTS AND ZERO-SHOT VOICE CONVERSION FOR EVERYONE

Título:	**YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone**
Autores:	**Edresson Casanova, Julian Weber, Christopher Shulby, Arnaldo Candido Junior, Eren Gölge, Moacir Antonelli Ponti**
Ano:	**2022**
Conferência:	**International Conference on Machine Learning (ICML)**
Situação:	**Publicado**

Motivação:

De acordo com Tan *et al.* (2021), a qualidade dos modelos de síntese de fala *zero-shot multi-speaker* atuais não é suficientemente boa, principalmente para locutores alvos com características de fala muito diferentes daquelas vistas em treinamento. Apesar do uso de *normalizing flows* ter alcançado resultados no estado da arte, a lacuna entre locutores vistos no treinamento e os novos ainda é uma questão de pesquisa em aberto. Além disso, os modelos de síntese de fala *zero-shot multi-speaker* ainda requerem um grande número de locutores durante o treinamento, dificultando a obtenção de modelos de alta qualidade em idiomas com poucos recursos. Embora o modelo SC-GlowTTS tenha alcançado resultados promissores com apenas 11 locutores, geralmente, limitar o número de locutores no treinamento dificulta ainda mais a generalização do modelo para locutores com características de fala muito diferentes daquelas vistas em treinamento. Por esses motivos, nesse trabalho explorou-se o uso de uma abordagem multilíngue, usufruindo da quantidade de locutores disponíveis em um idioma com muitos recursos disponíveis, e deste modo, mostrando a viabilidade do treinamento de um modelo de síntese de fala *zero-shot multi-speaker* com apenas um único locutor no idioma alvo.

Contribuições relevantes para a tese:

- Apresenta uma nova abordagem estado da arte para síntese de fala multilíngue *zero-shot multi-speaker*;
- A abordagem proposta permite a síntese de fala *zero-shot multi-speaker* e a conversão de voz *zero-shot* com qualidade e similaridade de fala promissora em um idioma alvo utilizando apenas um locutor nesse idioma;
- Para locutores com características de voz ou de gravação muito diferentes daquelas vistas no treinamento, o modelo proposto pode ser ajustado com menos de 1 minuto de fala e obter resultados no estados da arte em similaridade de voz, com qualidade promissora.

YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone

Edresson Casanova[1], *Julian Weber*[2], *Christopher Shulby*[3], *Arnaldo Candido Junior*[4], *Eren Gölge*[5] *and Moacir Antonelli Ponti*[1]

[1] Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Brazil
[2] Sopra Banking Software, France
[3] Defined.ai, United States of America
[4] Federal University of Technology – Paraná, Brazil
[5] Coqui, Germany
edresson@usp.br

Abstract

YourTTS brings the power of a multilingual approach to the task of zero-shot multi-speaker TTS. Our method builds upon the VITS model and adds several novel modifications for zero-shot multi-speaker and multilingual training. We achieved state-of-the-art (SOTA) results in zero-shot multi-speaker TTS and results comparable to SOTA in zero-shot voice conversion on the VCTK dataset. Additionally, our approach achieves promising results in a target language with a single-speaker dataset, opening possibilities for zero-shot multi-speaker TTS and zero-shot voice conversion systems in low-resource languages. Finally, it is possible to fine-tune the YourTTS model with less than 1 minute of speech and achieve state-of-the-art results in voice similarity and with reasonable quality. This is important to allow synthesis for speakers with a very different voice or recording characteristics from those seen during training.

Index Terms: cross-lingual zero-shot multi-speaker TTS, text-to-speech, cross-lingual zero-shot voice conversion, speaker adaptation.

1. Introduction

Text-to-Speech (TTS) systems have significantly advanced in recent years with deep learning approaches, allowing successful applications such as speech-based virtual assistants. Most TTS systems were tailored from a single speaker's voice, but there is current interest in synthesizing voices for new speakers (not seen during training), employing only a few seconds of speech. This approach is called zero-shot multi-speaker TTS (ZS-TTS) as in [1, 2, 3, 4].

ZS-TTS using deep learning was first proposed by [5] which extended the DeepVoice 3 method [6]. Meanwhile, Tacotron 2 [7] was adapted using external speaker embeddings extracted from a trained speaker encoder using a generalized end-to-end loss (GE2E) [8], allowing for speech generation that resembles the target speaker [1]. Similarly, Tacotron 2 was used with a different speaker embeddings methods [2], with LDE embeddings [9] to improve similarity and naturalness of speech for unseen speakers [10]. The authors also showed that a gender-dependent model improves the similarity for unseen speakers [2]. In this context, Attentron [3] proposed a fine-grained encoder with an attention mechanism for extracting detailed styles from various reference samples and a coarse-grained encoder. As a result of using several reference samples, they achieved better voice similarity for unseen speakers. ZSM-SS [11] is a Transformer-based architecture with a normalization architecture and an external speaker encoder based on Wav2vec 2.0 [12]. The authors conditioned the normalization architecture with speaker embeddings, pitch, and energy. Despite promising results, the authors did not compare the proposed model with any of the related works mentioned above. SC-GlowTTS [4] was the first application of flow-based models in ZS-TTS. It improved voice similarity for unseen speakers in training with respect to previous studies while maintaining comparable quality.

Despite these advances, the similarity gap between observed and unobserved speakers during training is still an open research question. ZS-TTS models still require a considerable amount of speakers for training, making it difficult to obtain high-quality models in low-resource languages. Furthermore, according to [13], the quality of current ZS-TTS models is not sufficiently good, especially for target speakers with speech characteristics that differ from those seen in training. Although SC-GlowTTS [4] achieved promising results with only 11 speakers from the VCTK dataset [14], when one limits the number and variety of training speakers, it also further hinders the model generalization for unseen voices.

In parallel with the ZS-TTS, multilingual TTS has also evolved aiming at learning models for multiple languages at the same time [15, 16, 17, 18]. Some of these models are particularly interesting as they allow for code-switching, i.e. changing the target language for some part of a sentence, while keeping the same voice [17]. This can be useful in ZS-TTS as it allows using of speakers from one language to be synthesized in another language.

In this paper, we propose YourTTS with several novel ideas focused on zero-shot multi-speaker and multilingual training. We report state-of-the-art zero-shot multi-speaker TTS results, as well as results comparable to SOTA in zero-shot voice conversion for the VCTK dataset.

Our novel zero-shot multi-speaker TTS approach includes the following contributions:

- State-of-the-art results in the English Language;
- The first work proposing a multilingual approach in the zero-shot multi-speaker TTS scope;
- Ability to do zero-shot multi-speaker TTS and zero-shot Voice Conversion with promising quality and similarity in a target language using only one speaker in the target language during model training;

- Require less than 1 minute of speech to fine-tune the model for speakers who have voice/recording characteristics very different from those seen in model training, and still achieve good similarity and quality results.

The audio samples for each of our experiments are available on the demo web-site[1]. For reproducibility, our source-code is available at the Coqui TTS[2], as well as the model checkpoints of all experiments[3].

2. YourTTS Model

YourTTS builds upon VITS [19], but includes several novel modifications for zero-shot multi-speaker and multilingual training. First, unlike previous work [4, 19], in our model we used raw text as input instead of phonemes. This allows more realistic results for languages without good open-source grapheme-to-phoneme converters available.

As in previous works, e.g. [19], we use a transformer-based text encoder [20, 4]. However, for multilingual training, we concatenate 4-dimensional trainable language embeddings into the embeddings of each input character. In addition, we also increased the number of transformer blocks to 10 and the number of hidden channels to 196. As a decoder, we use a stack of 4 affine coupling layers [21] each layer is itself a stack of 4 WaveNet residual blocks [22], as in VITS model.

As a vocoder we use the HiFi-GAN [23] version 1 with the discriminator modifications introduced by [19]. Furthermore, for efficient end2end training, we connect the TTS model with the vocoder using a variational autoencoder (VAE) [24]. For this, we use the Posterior Encoder proposed by [19]. The Posterior Encoder consists of 16 non-causal WaveNet residual blocks [25, 20]. As input, the Posterior Encoder receives a linear spectrogram and predicts a latent variable, this latent variable is used as input for the vocoder and for the flow-based decoder, thus, no intermediate representation (such as mel-spectrograms) is necessary. This allows the model to learn an intermediate representation; hence, it achieves superior results to a two-stage approach system in which the vocoder and the TTS model are trained separately [19]. Furthermore, to enable our model to synthesize speech with diverse rhythms from the input text, we use the stochastic duration predictor proposed in [19].

YourTTS during training and inference is illustrated in Figure 1, where $(+\!\!\!\!+)$ indicates concatenation, red connections mean no gradient will be propagated by this connection, and dashed connections are optional. We omit the Hifi-GAN discriminator networks for simplicity.

To give the model zero-shot multi-speaker generation capabilities we condition all affine coupling layers of the flow-based decoder, the posterior encoder, and the vocoder on external speaker embeddings. We use global conditioning [22] in the residual blocks of the coupling layers as well as in the posterior encoder. We also sum the external speaker embeddings with the text encoder output and the decoder output before we pass them to the duration predictor and the vocoder, respectively. We use linear projection layers to match the dimensions before element-wise summations (see Figure 1).

Also, inspired by [26], we investigated Speaker Consistency Loss (SCL) in the final loss. In this case, a pre-trained speaker encoder is used to extract speaker embeddings from the generated audio and ground truth on which we maximize the cosine similarity. Formally, let $\phi(.)$ be a function outputting the embedding of a speaker, cos_sim be the cosine similarity function, α a positive real number that controls the influence of the SCL in the final loss, and n the batch size, the SCL is defined as follows:

$$L_{SCL} = \frac{-\alpha}{n} \cdot \sum_{i}^{n} cos\_sim(\phi(g_i), \phi(h_i)), \tag{1}$$

where g and h represent, respectively, the ground truth and the generated speaker audio.

During training, the Posterior Encoder receives linear spectrograms and speaker embeddings as input and predicts a latent variable z. This latent variable and speaker embeddings are used as input to the GAN-based vocoder generator which generates the waveform. For efficient end-to-end vocoder training, we randomly sample constant length partial sequences from z as in [23, 27, 28, 19]. The Flow-based decoder aims to condition the latent variable z and speaker embeddings with respect to a P_{Zp} prior distribution. To align the P_{Zp} distribution with the output of the text encoder, we use the Monotonic Alignment Search (MAS) [20, 19]. The stochastic duration predictor receives as input speaker embeddings, language embeddings and the duration obtained through MAS. To generate human-like rhythms of speech, the objective of the stochastic duration predictor is a variational lower bound of the log-likelihood of the phoneme (pseudo-phoneme in our case) duration.

During inference, MAS is not used. Instead, P_{Zp} distribution is predicted by the text encoder and the duration is sampled from random noise through the inverse transformation of the stochastic duration predictor and then, converted to integer. In this way, a latent variable z_p is sampled from the distribution P_{Zp}. The inverted Flow-based decoder receives as input the latent variable z_p and the speaker embeddings, transforming the latent variable z_p into the latent variable z which is passed as input to the vocoder generator, thus obtaining the synthesized waveform.

3. Experiments

3.1. Speaker Encoder

As speaker encoder, we use the H/ASP model [29] publicly available, that was trained with the Prototypical Angular [30] plus Softmax loss functions in the VoxCeleb 2 [31] dataset. This model was chosen for achieving state-of-the-art results in VoxCeleb 1 [32] test subset. In addition, we evaluated the model in the test subset of Multilingual LibriSpeech (MLS) [33] using all languages. This model reached an average Equal Error Rate (EER) of 1.967 while the speaker encoder used in the SC-GlowTTS paper [4] reached an EER of 5.244.

3.2. Audio datasets

We investigated 3 languages, using one dataset per language to train the model. For all datasets, pre-processing was carried out in order to have samples of similar loudness and to remove long periods of silence. All the audios to 16Khz and applied voice activity detection (VAD) using Webrtcvad toolkit[4] to trim the trailing silences. Additionally, we normalized all audio to -27dB using the RMS-based normalization from the Python package ffmpeg-normalize[5].

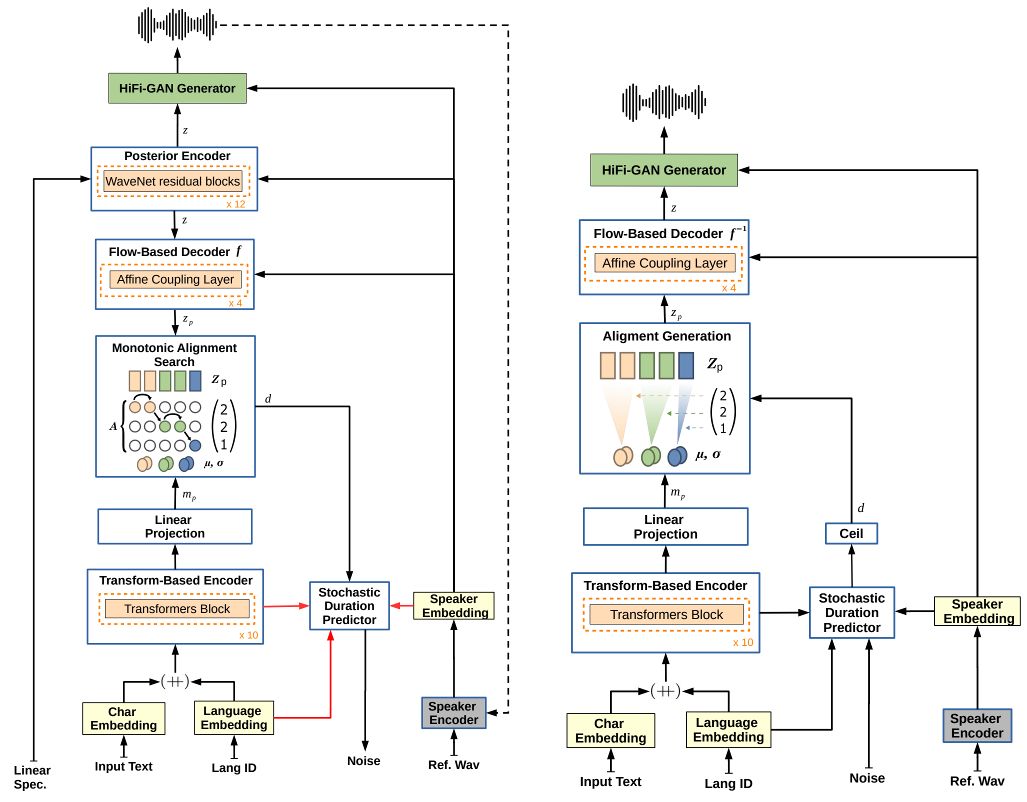
[1] https://edresson.github.io/YourTTS/

[2] https://github.com/coqui-ai/TTS

[3] https://github.com/Edresson/YourTTS

[4] https://github.com/wiseman/py-webrtcvad

[5] https://github.com/slhck/ffmpeg-normalize



(a) Training procedure

(b) Inference procedure

Figure 1: *YourTTS diagram depicting (a) training procedure and (b) inference procedure.*

English: VCTK [14] dataset, which contains 44 hours of speech and 109 speakers, sampled at 48KHz. We divided the VCTK dataset into: train, development (containing the same speakers as the train set) and test. For the test set, we selected 11 speakers that are neither in the development nor the training set; following the proposal by [1] and [4], we selected 1 representative from each accent totaling 7 women and 4 men (speakers 225, 234, 238, 245, 248, 261, 294, 302, 326, 335 and 347). Furthermore, in some experiments we used the subsets *train-clean-100* and *train-clean-360* of the LibriTTS dataset [34] seeking to increase the number of speakers in the training of the models.

Portuguese: TTS-Portuguese Corpus [35], a single-speaker dataset of the Brazilian Portuguese language with around 10 hours of speech, sampled at 48KHz. As the authors did not use a studio, the dataset contains ambient noise. We used the FullSubNet model [36] as denoiser and resampled the data to 16KHz. For development we randomly selected 500 samples and the rest of the dataset was used for training.

French: fr_FR set of the M-AILABS dataset [37], which is based on LibriVox[6]. It consists of 2 female (104h) and 3 male speakers (71h) sampled at 16KHz.

To evaluate the zero-shot multi-speaker capabilities of our model in English, we use the 11 VCTK speakers reserved for testing. To further test its performance outside of the VCTK domain, we select 10 speakers (5F/5M) from subset *test-clean* of LibriTTS dataset [34]. For Portuguese we select samples from 10 speakers (5F/5M) from the Multilingual LibriSpeech (MLS) [33] dataset. For French, no evaluation dataset was used, due to the reasons described in Section 4. Finally, for speaker adaptation experiments, to mimic a more realistic setting, we used 4 speakers from the Common Voice dataset [38].

3.3. Experimental setup

We carried out four training experiments with YourTTS:

- **Experiment 1:** using VCTK dataset (monolingual);
- **Experiment 2:** using both VCTK and TTS-Portuguese datasets (bilingual);
- **Experiment 3:** using VCTK, TTS-Portuguese and M-AILABS french datasets (trilingual);
- **Experiment 4:** starting with the model obtained in experiment 3 we continue training with 1151 additional English speakers from both LibriTTS partitions *train-clean-100* and *train-clean-360*.

To accelerate training, in every experiment, we use transfer learning. In experiment 1, we start from a model trained 1M steps on LJSpeech [39] and continue the training for 200K steps with the VCTK dataset. However, due to the proposed changes, some layers of the model were randomly initialized due to the incompatibility of the shape of the weights. For experiments 2 and 3, training is done by continuing from the previous experiment for approximately 140k steps, learning one language at a time. In addition, for each of the experiments a fine-tuning was

[6] https://librivox.org/

performed for 50k steps using the Speaker Consistency Loss (SCL), described in section 2, with $\alpha = 9$. Finally, for experiment 4, we continue training from the model from experiment 3 fine-tuned with the Speaker Consistency Loss. Note that, although the latest works in ZS-TTS [2, 3, 4] only use the VCTK dataset, this dataset has a limited number of speakers (109) and little variety of recording conditions. Thus, after training with VCTK only, in general, ZS-TTS models do not generalize satisfactorily to new speakers where recording conditions or voice characteristics are very different than those seen in the training [13].

The models were trained using an NVIDIA TESLA V100 32GB with a batch size of 64. For the TTS model training and for the discrimination of vocoder HiFi-GAN we use the AdamW optimizer [40] with betas 0.8 and 0.99, weight decay 0.01 and an initial learning rate of 0.0002 decaying exponentially by a gamma of 0.999875 [41]. For the multilingual experiments, we use weighted random sampling [41] to guarantee a language balanced batch.

4. Results and Discussion

In this paper, we evaluate synthesized speech quality using a Mean Opinion Score (MOS) study, as in [42]. To compare the similarity between the synthesized voice and the original speaker, we calculate the Speaker Encoder Cosine Similarity (SECS) [4] between the speaker embeddings of two audios extracted from the speaker encoder. It ranges from -1 to 1, and a larger value indicates a stronger similarity [2]. Following previous works [3, 4], we compute SECS using the speaker encoder of the Resemblyzer [43] package, allowing for comparison with those studies. We also report the Similarity MOS (Sim-MOS) following the works of [1], [3], and [4].

Although the experiments involve 3 languages, due to the high cost of the MOS metrics, only two languages were used to compute such metrics: English, which has the largest number of speakers, and Portuguese, which has the smallest number. In addition, following the work of [4] we present such metrics only for speakers unseen during training.

MOS scores were obtained with rigorous crowdsourcing[7]. For the calculation of MOS and the Sim-MOS in the English language, we use 276 and 200 native English contributors, respectively. For the Portuguese language, we use 90 native Portuguese contributors for both metrics.

During evaluation we use the fifth sentence of the VCTK dataset (i.e, speakerID_005.txt) as reference audio for the extraction of speaker embeddings, since all test speakers uttered it and because it is a long sentence (20 words). For the LibriTTS and MLS Portuguese, we randomly draw one sample per speaker considering only those with 5 seconds or more, to guarantee a reference with sufficient duration.

For the calculation of MOS, SECS, and Sim-MOS in English, we select 55 sentences randomly from the *test-clean* subset of the LibriTTS dataset, considering only sentences with more than 20 words. For Portuguese we use the translation of these 55 sentences. During inference, we synthesize 5 sentences per speaker in order to ensure coverage of all speakers and a good number of sentences. As ground truth for all test subsets, we randomly select 5 audios for each of the test speakers. For the SECS and Sim-MOS ground truth, we compared such randomly selected 5 audios per speaker with the reference audios used for the extraction of speaker embeddings during synthesis of the test sentences.

Table 1 shows MOS and Sim-MOS with 95% confidence intervals and SECS for all of our experiments in English for the datasets VCTK and LibriTTS and in Portuguese with the Portuguese sub-set of the dataset MLS.

4.1. VCTK dataset

For the VCTK dataset, the best similarity results were obtained with experiments 1 (monolingual) and 2 + SCL (bilingual). Both achieved the same SECS and a similar Sim-MOS. According to the Sim-MOS, the use of SCL did not bring any improvements; however, the confidence intervals of all experiments overlap, making this analysis inconclusive. On the other hand, according to SECS, using SCL improved the similarity in 2 out of 3 experiments. Also, for experiment 2, both metrics agree on the positive effect of SCL in similarity.

Another noteworthy result is that SECS for all of our experiments on the VCTK dataset are higher than the ground truth. This can be explained by characteristics of the VCTK dataset itself which has, for example, significant breathing sounds in most audios. The speaker encoder may not be able to handle these features, hereby lowering the SECS of the ground truth. Overall, in our best experiments with VCTK, the similarity (SECS and Sim-MOS) and quality (MOS) results are similar to the ground truth. Our results in terms of MOS match the ones reported by the VITS article [19]. However, we show that with our modifications, the model manages to maintain good quality and similarity for unseen speakers. Finally, our best experiments achieve superior results in similarity and quality when compared to [3, 4]; therefore, achieving the SOTA in the VCTK dataset for zero-shot multi-speaker TTS.

4.2. LibriTTS dataset

We achieved the best LibriTTS similarity in experiment 4. This result can be explained by the use of more speakers ($\sim$ 1.2k) than any other experiments ensuring a broader coverage of voice and recording condition diversity. On the other hand, MOS achieved the best result for the monolingual case. We believe that this was mainly due to the quality of the training datasets. Experiment 1 uses VCTK dataset only, which has higher quality when compared to other datasets added in the other experiments.

4.3. Portuguese MLS dataset

For the Portuguese MLS dataset, the highest MOS metric was achieved by experiment 3+SCL, with MOS 4.11±0.07, although the confidence intervals overlap with the other experiments. It is interesting to observe that the model trained in Portuguese with a single-speaker dataset of medium quality, manages to reach a good quality in the zero-shot multi-speaker synthesis. Experiment 3 is the best experiment according to Sim-MOS (3.19±0.10) however, with an overlap with other ones considering the confidence intervals. In this dataset, Sim-MOS and SECS do not agree: based on the SECS metric, the model with higher similarity was obtained in experiment 4+SCL. We believe this is due to the variety in the LibriTTS dataset. The dataset is also composed of audiobooks, therefore tending to have similar recording characteristics and prosody to the MLS dataset. We believe that this difference between SECS and Sim-MOS can be explained by the confidence intervals of Sim-MOS. Finally, Sim-MOS achieved in this dataset is relevant, considering that our model was trained with only one male speaker in

[7] https://www.definedcrowd.com/evaluation-of-experience/

Table 1: *SECS, MOS and Sim-MOS with 95% confidence intervals for all our experiments.*

	VCTK			LibriTTS			MLS-PT		
Exp.	SECS	MOS	Sim-MOS	SECS	MOS	Sim-MOS	SECS	MOS	Sim-MOS
Ground Truth	0.824	4.26±0.04	4.19±0.06	0.931	4.22±0.05	4.22±0.06	0.9018	4.61±0.05	4.41±0.05
Attentron ZS	(0.731)	(3.86±0.05)	(3.30 ±0.06)	–	–	–	–	–	–
SC-GlowTTS	(0.804)	(3.78±0.07)	(3.99±0.07)	–	–	–	–	–	–
Exp. 1	**0.864**	4.21±0.04	4.16±0.05	0.754	**4.25±0.05**	3.98±0.07	–	–	–
Exp. 1 + SCL	0.861	4.20±0.05	4.13±0.06	0.765	4.21±0.04	4.05±0.07	–	–	–
Exp. 2	0.857	**4.24±0.04**	4.15±0.06	0.762	4.22±0.05	4.01±0.07	0.740	3.96±0.08	3.02±0.1
Exp. 2 + SCL	**0.864**	4.19±0.05	**4.17±0.06**	0.773	4.23±0.05	4.01±0.07	0.745	4.09±0.07	2.98±0.1
Exp. 3	0.851	4.21±0.04	4.10±0.06	0.761	4.21±0.04	4.01±0.05	0.761	4.01±0.08	**3.19±0.1**
Exp. 3 + SCL	0.855	4.22±0.05	4.06±0.06	0.778	4.17±0.05	3.98±0.07	0.766	**4.11±0.07**	3.17±0.1
Exp. 4 + SCL	0.843	4.23±0.05	4.10±0.06	**0.856**	4.18±0.05	**4.07±0.07**	**0.798**	3.97±0.08	3.07±0.1

the Portuguese language.

Analyzing the metrics by **gender**, the MOS for experiment 4 considering only male and female speakers are respectively 4.14 ± 0.11 and 3.79 ± 0.12. Also, the Sim-MOS for male and female speakers are respectively 3.29 ± 0.14 and 2.84 ± 0.14. Therefore, the performance of our model in Portuguese is affected by gender. We believe that this happened because our model was not trained with female Portuguese speakers. Despite that, our model was able to produce female speech in the Portuguese language. The Attentron model achieved a Sim-MOS of 3.30±0.06 after being trained with approximately 100 speakers in the English language. Considering confidence intervals, our model achieved a similar Sim-MOS even when seeing only one male speaker in the target language. Hence, we believe that our approach can be the solution for the development of zero-shot multi-speaker TTS models in low-resourced languages.

Including **French** (i.e. experiment 3) appear to have improved both quality and similarity (according to SECS) in Portuguese. The increase in quality can be explained by the fact that the M-AILABS French dataset has better quality than the Portuguese corpus; consequently, as the batch is balanced by language, there is a decrease in the amount of lower quality speech in the batch during model training. Also, increase in similarity can be explained by the fact that TTS-Portuguese is a single speaker dataset and with the batch balancing by language in experiment 2, half of the batch is composed of only one male speaker. When French is added, then only a third of the batch will be composed of the Portuguese speaker voice.

4.4. Speaker Consistency Loss

The use of Speaker Consistency Loss (SCL) improved similarity measured by SECS. On the other hand, for the Sim-MOS the confidence intervals between the experiments are inconclusive to assert that the SCL improves similarity. Nevertheless, we believe that SCL can help the generalization in recording characteristics not seen in training. For example, in experiment 1, the model did not see the recording characteristics of the LibriTTS dataset in training but during testing on this dataset, both the SECS and Sim-MOS metrics showed an improvement in similarity thanks to SCL. On the other hand, it seems that using SCL slightly decreases the quality of generated audio. We believe this is because with the use of SCL, our model learns to generate recording characteristics present in the reference audio, producing more distortion and noise. However, it should be noted that in our tests with high-quality reference samples, the model is able to generate high-quality speech.

5. Zero-Shot Voice Conversion

As in the SC-GlowTTS [4] model, we do not provide any information about the speaker's identity to the encoder, so the distribution predicted by the encoder is forced to be speaker independent. Therefore, YourTTS can convert voices using the model's Posterior Encoder, decoder and the HiFi-GAN Generator. Since we conditioned YourTTS with external speaker embeddings, it enables our model to mimic the voice of unseen speakers in a zero-shot voice conversion setting.

In [44], the authors reported the MOS and Sim-MOS metrics for the AutoVC [45] and NoiseVC [44] models for 10 VCTK speakers not seen during training. To compare our results, we selected 8 speakers (4M/4F) from the VCTK test subset. Although [44] uses 10 speakers, due to gender balance, we were forced to use only 8 speakers.

Furthermore, to analyze the generalization of the model for the Portuguese language, and to verify the result achieved by our model in a language where the model was trained with only one speaker, we used the 8 speakers (4M/4F) from the test subset of the MLS Portuguese dataset. Therefore, in both languages we use speakers not seen in the training. Following [45] for a deeper analysis, we compared the transfer between male, female and mixed gender speakers individually. During the analysis, for each speaker, we generate a transfer in the voice of each of the other speakers, choosing the reference samples randomly, considering only samples longer than 3 seconds. In addition, we analyzed voice transfer between English and Portuguese speakers. We calculate the MOS and the Sim-MOS as described in Section 4. However, for the calculation of the sim-MOS when transferring between English and Portuguese (pt-en and en-pt), as the reference samples are in one language and the transfer is done in another language, we used evaluators from both languages (58 and 40, respectively, for English and Portuguese).

Table 2 presents the MOS and Sim-MOS for these experiments. Samples of the zero-shot voice conversion are present in the demo page[8].

5.1. Intra-lingual results

For zero-shot voice conversion from one English-speaker to another English-speaker (en-en) our model achieved a MOS of 4.20±0.05 and a Sim-MOS of 4.07±0.06. For comparison in [44] the authors reported the MOS and Sim-MOS results for the AutoVC [45] and NoiseVC [44] models. For 10 VCTK speakers not seen during training, the AutoVC model achieved

[8] https://edresson.github.io/YourTTS/

Table 2: *MOS and Sim-MOS with 95% confidence intervals for the zero-shot voice conversion experiments.*

REF/TAR	M-M		M-F		F-F		F-M		ALL	
	MOS	SIM-MOS	MOS	SIM-MOS	MOS	SIM-MOS	MOS	SIM-MOS	MOS	SIM-MOS
EN-EN	4.22±0.10	4.15±0.12	4.14±0.09	4.11±0.12	4.16±0.12	3.96±0.15	4.26±0.09	4.05±0.11	4.20±0.05	4.07±0.06
PT-PT	3.84 ± 0.18	3.80 ± 0.15	3.46 ± 0.10	3.12 ± 0.17	3.66 ± 0.2	3.35 ± 0.19	3.67 ± 0.16	3.54 ± 0.16	3.64 ± 0.09	3.43 ± 0.09
EN-PT	4.17±0.09	3.68 ± 0.10	4.24±0.08	3.54 ± 0.11	4.14±0.09	3.58 ± 0.12	4.12±0.10	3.58 ± 0.11	4.17±0.04	3.59 ± 0.05
PT-EN	3.62 ± 0.16	3.8 ± 0.10	2.95 ± 0.2	3.67 ± 0.11	3.51 ± 0.18	3.63 ± 0.11	3.47 ± 0.18	3.57 ± 0.11	3.40 ± 0.09	3.67 ± 0.05

a MOS of 3.54 ± 1.08[9] and a Sim-MOS of 1.91 ± 1.34. On the other hand, the NoiseVC model achieved a MOS of 3.38 ± 1.35 and a Sim-MOS of 3.05 ± 1.25. Therefore, our model achieved results comparable to the SOTA in zero-shot voice conversion in the VCTK dataset. Alhtough the model was trained with more data and speakers, the similarity results of the VCTK dataset in Section 4 indicate that the model trained with only the VCTK dataset (experiment 1) presents a better similarity than the model explored in this Section (experiment 4). Therefore, we believe that YourTTS can achieve a result very similar or even superior in zero-shot voice conversion when being trained and evaluated using only the VCTK dataset.

For zero-shot voice conversion from one Portuguese speaker to another Portuguese speaker our model achieved a MOS of 3.64 ± 0.09 and a Sim-MOS of 3.43 ± 0.09. We note that our model performs significantly worse in voice transfer similarity between female speakers (3.35 ± 0.19) compared to transfers between male speakers (3.80 ± 0.15). This can be explained by the lack of female speakers for the Portuguese language during the training of our model. Again, it is remarkable that our model manages to approximate female voices in Portuguese without ever having seen a female voice in that language.

5.2. Cross-lingual results

Apparently, the transfer between English and Portuguese speakers works as well as the transfer between Portuguese speakers. However, for the transfer of a Portuguese speaker to an English speaker (pt-en) the MOS scores drop in quality. This was especially due to the low quality of voice conversion from Portuguese male speakers to English female speakers. In general, as discussed above, due to the lack of female speakers in the training of the model, the transfer to female speakers achieves poor results. In this case, the challenge is even greater as it is necessary to convert audios from a male speaker in Portuguese to the voice of a English female speaker.

In English, during conversions, the speaker's gender did not significantly influence the model's performance. However, for transfers involving Portuguese, the absence of female voices in the training of the model hindered generalization.

6. Speaker Adaptation

The different recording conditions are a challenge for the generalization of the zero-shot multi-speaker TTS models. Speakers who have a voice that differs greatly from those seen in training also become a challenge [13]. Nevertheless, to show the potential of our model for adaptation to new speakers/recording conditions, we selected samples from 20 to 61 seconds of speech for 2 Portuguese and 2 English speakers (1M/1F) in the Common Voice [38] dataset. Using these 4 speakers, we perform fine-tuning on the checkpoint from experiment 4 with Speaker Consistency Loss individually for each speaker.

During fine-tuning, to ensure that multilingual synthesis is not impaired, we use all the datasets used in experiment 4. However, we use Weighted random sampling [41] to guarantee that samples from adapted speakers appear in a quarter of the batch. The model is trained that way for 1500 steps. For evaluation, we use the same approach described in Section 4.

Table 3 shows the gender, total duration in seconds and number of samples used during the training for each speaker, and the metrics SECS, MOS and Sim-MOS for the ground truth (GT), zero-shot multi-speaker TTS mode (ZS), and the fine-tuning (FT) with speaker samples.

In general, our model's fine-tuning with less than 1 minute of speech from speakers who have recording characteristics not seen during training achieved very promising results, significantly improving similarity in all experiments.

In English, the results of our model in zero-shot multi-speaker TTS mode are already good and after fine-tuning both male and female speakers achieved Sim-MOS comparable to the ground truth. The fine-tuned model achieves greater SECS than the ground truth, which was already observed in previous experiments. We believe that this phenomenon can be explained by the model learning to copy the recording characteristics and reference sample's distortions, giving an advantage over other real speaker samples.

In Portuguese, compared to zero-shot, fine-tuning seems to trade a bit of naturalness for a much better similarity. For the male speaker, the Sim-MOS increased from 3.35±0.12 to 4.19±0.07 after fine-tuning with just 31 seconds of speech for that speaker. For the female speaker, the similarity improvement was even more impressive, going from 2.77±0.15 in zero-shot mode to 4.43±0.06 after the fine-tuning with just 20 seconds of speech from that speaker.

Although our model manages to achieve high similarity using only seconds of the target speaker's speech, Table 3 seems to presents a direct relationship between the amount of speech used and the naturalness of speech (MOS). With approximately 1 minute of speech in the speaker's voice our model can copy the speaker's speech characteristics, even increasing the naturalness compared to zero-shot mode. On the other hand, using 44 seconds or less of speech reduces the quality/naturalness of the generated speech when compared to the zero-shot or ground truth model. Therefore, although our model shows good results in copying the speaker's speech characteristics using only 20 seconds of speech, more than 45 seconds of speech are more adequate to allow higher quality. Finally, we also noticed that voice conversion improves significantly after fine-tuning the model, mainly in Portuguese and French where few speakers are used in training.

[9] The authors presented the results in a graph without the actual figures, so the MOS scores reported here are approximations calculated considering the length in pixels of those graphs.

Table 3: *SECS, MOS and Sim-MOS with 95% confidence intervals for the speaker adaptation experiments.*

	Sex	Dur. (Sam.)	Mode	SECS	MOS	Sim-MOS
EN	M	61s (15)	GT	0.875	4.17±0.09	**4.08±0.13**
			ZS	0.851	4.11±0.07	4.04±0.09
			FT	**0.880**	4.17±0.07	**4.08±0.09**
	F	44s (11)	GT	0.894	4.25±0.11	**4.17±0.13**
			ZS	0.814	4.12±0.08	4.11±0.08
			FT	**0.896**	4.10±0.08	**4.17±0.08**
PT	M	31s (7)	GT	0.880	4.76±0.12	**4.31±0.14**
			ZS	0.817	4.03±0.11	3.35±0.12
			FT	**0.915**	3.74±0.12	4.19±0.07
	F	20s (5)	GT	0.873	4.62±0.19	**4.65±0.14**
			ZS	0.743	3.59±0.13	2.77±0.15
			FT	**0.930**	3.48±0.13	4.43±0.06

7. Conclusions, limitations and future work

In this work, we presented YourTTS, which achieved SOTA results in zero-shot multi-speaker TTS and zero-shot voice conversion in the VCTK dataset. Furthermore, we show that our model can achieve promising results in a target language using only a single speaker dataset. Additionally, we show that for speakers who have both a voice and recording conditions that differ greatly from those seen in training, our model can be adjusted to a new voice using less than 1 minute of speech.

However, our model exhibits some limitations. For the TTS experiments in all languages, our model presents instability in the stochastic duration predictor which, for some speakers and sentences, generates unnatural durations. We also note that mispronunciations occur for some words, especially in Portuguese. Unlike [35, 46, 19], we do not use phonetic transcriptions, making our model more prone to such problems. For Portuguese voice conversion, the speaker's gender significantly influences the model's performance, due to the absence of female voices in training. For Speaker Adaptation, although our model shows good results in copying the speaker's speech characteristics using only 20 seconds of speech, more than 45 seconds of speech are more adequate to allow higher quality.

In future work, we intend to seek improvements to the duration predictor of the YourTTS model as well as training in more languages. Furthermore, we intend to explore the application of this model for data augmentation in the training of automatic speech recognition models in low-resource settings.

8. Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001, as well as CNPq (National Council of Technological and Scientific Development) grants 304266/2020-5. In addition, this research was financed in part by Artificial Intelligence Excellence Center (CEIA)[10] via projects funded by the Department of Higher Education of the Ministry of Education (SESU/MEC) and Cyberlabs Group[11]. Also, we would like to thank the Defined.ai[12] for making industrial-level MOS testing so easily available. Finally, we would like to thank all contributors to the Coqui TTS repository[13], this work was only possible thanks to the commitment of all.

9. References

[1] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.

[2] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6184–6188.

[3] S. Choi, S. Han, D. Kim, and S. Ha, "Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding," *arXiv preprint arXiv:2005.08484*, 2020.

[4] E. Casanova, C. Shulby, E. Gölge, N. M. Müller, F. S. de Oliveira, A. Candido Jr., A. da Silva Soares, S. M. Aluisio, and M. A. Ponti, "SC-GlowTTS: An Efficient Zero-Shot Multi-Speaker Text-To-Speech Model," in *Proc. Interspeech 2021*, 2021, pp. 3645–3649.

[5] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 019–10 029.

[6] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," *arXiv preprint arXiv:1710.07654*, 2017.

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

[8] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[9] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," *arXiv preprint arXiv:1804.05160*, 2018.

[10] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[10] http://centrodeia.org

[11] https://cyberlabs.ai

[12] https://www.defined.ai

[13] https://github.com/coqui-ai/TTS

[11] N. Kumar, S. Goel, A. Narang, and B. Lall, "Normalization Driven Zero-Shot Multi-Speaker Speech Synthesis," in *Proc. Interspeech 2021*, 2021, pp. 1354–1358.

[12] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[13] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *arXiv preprint arXiv:2106.15561*, 2021.

[14] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2016.

[15] Y. Cao, X. Wu, S. Liu, J. Yu, X. Li, Z. Wu, X. Liu, and H. Meng, "End-to-end code-switched tts with mix of monolingual recordings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6935–6939.

[16] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," *Proc. Interspeech 2019*, pp. 2080–2084, 2019.

[17] T. Nekvinda and O. Dušek, "One model, many languages: Meta-learning for multilingual text-to-speech," *Proc. Interspeech 2020*, pp. 2972–2976, 2020.

[18] S. Li, B. Ouyang, L. Li, and Q. Hong, "Light-tts: Lightweight multi-speaker multi-lingual text-to-speech," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 8383–8387.

[19] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," *arXiv preprint arXiv:2106.06103*, 2021.

[20] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *arXiv preprint arXiv:2005.11129*, 2020.

[21] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=HkpbnH9lx

[22] A. v. d. 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," *arXiv preprint arXiv:1609.03499*, 2016.

[23] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *arXiv preprint arXiv:2010.05646*, 2020.

[24] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[25] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3617–3621.

[26] D. Xin, Y. Saito, S. Takamichi, T. Koriyama, and H. Saruwatari, "Cross-Lingual Speaker Adaptation Using Domain Adaptation and Speaker Consistency Loss for Text-To-Speech Synthesis," in *Proc. Interspeech 2021*, 2021, pp. 1614–1618.

[27] M. Bińkowski, J. Donahue, S. Dieleman, A. Clark, E. Elsen, N. Casagrande, L. C. Cobo, and K. Simonyan, "High fidelity speech synthesis with adversarial networks," in *International Conference on Learning Representations*, 2019.

[28] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2021.

[29] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova baseline system for the voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2009.14153*, 2020.

[30] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," in *Interspeech*, 2020.

[31] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech 2018*, 2018, pp. 1086–1090. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1929

[32] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[33] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "MLS: A large-scale multilingual dataset for speech research," in *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, H. Meng, B. Xu, and T. F. Zheng, Eds. ISCA, 2020, pp. 2757–2761. [Online]. Available: https://doi.org/10.21437/Interspeech.2020-2826

[34] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.

[35] E. Casanova, A. C. Junior, C. Shulby, F. S. de Oliveira, J. P. Teixeira, M. A. Ponti, and S. M. Aluisio, "Tts-portuguese corpus: a corpus for speech synthesis in brazilian portuguese," 2020.

[36] X. Hao, X. Su, R. Horaud, and X. Li, "Fullsubnet: A full-band and sub-band fusion model for real-time single-channel speech enhancement," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun 2021. [Online]. Available: http://dx.doi.org/10.1109/ICASSP39728.2021.9414177

[37] Munich Artificial Intelligence Laboratories GmbH, "The m-ailabs speech dataset – caito," 2017. [Online]. Available: https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/

[38] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 4218–4222.

[39] K. Ito *et al.*, "The lj speech dataset," 2017.

[40] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017.

[41] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, pp. 8026–8037, 2019.

[42] F. Ribeiro, D. Florêncio, C. Zhang, and M. Seltzer, "Crowdmos: An approach for crowdsourcing mean opinion score studies," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2416–2419.

[43] C. Jemine, "Master thesis: Real-time voice cloning," 2019.

[44] S. Wang and D. Borth, "Noisevc: Towards high quality zero-shot voice conversion," *arXiv preprint arXiv:2104.06074*, 2021.

[45] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5210–5219.

[46] E. Casanova, A. C. Junior, F. S. de Oliveira, C. Shulby, J. P. Teixeira, M. A. Ponti, and S. M. Aluisio, "End-to-end speech synthesis applied to brazilian portuguese," *arXiv preprint arXiv:2005.05144*, 2020.

CAPÍTULO 7

A SINGLE SPEAKER IS ALMOST ALL YOU NEED FOR AUTOMATIC SPEECH RECOGNITION

Título:	**A single speaker is almost all you need for automatic speech recognition**
Autores:	**Edresson Casanova, Christopher Shulby, Arnaldo Candido Junior, Sandra Aluísio, Moacir Antonelli Ponti**
Ano:	**2022**
Revista:	**IEEE Signal Processing Letters**
Situação:	**Submetido - Sob revisão**

Motivação:

Embora trabalhos na literatura mostrem o potencial de modelos de síntese de fala *multi-speaker* ou *zero-shot multi-speaker* no aumento de dados para o treinamento de modelos ASR, melhorando o desempenho desses sistemas, os modelos até então explorados na literatura exigiam *datasets* de alta qualidade com muitos locutores e horas de fala para convergirem (LAPTEV *et al.*, 2020). No entanto, a grande maioria dos idiomas não possui um *dataset* de síntese de fala aberto e os poucos *datasets* disponíveis possuem um número limitado de locutores disponíveis. Deste modo, essa tecnologia era utilizada para idiomas com muitos recursos disponíveis. Nesse trabalho, exploramos o uso de um modelo de síntese de fala *zero-shot multi-speaker* treinado com apenas um único locutor em dois idiomas alvos, mostrando a viabilidade de aplicação desse método para idiomas com apenas um único locutor disponível.

Contribuições relevantes para a tese:

- Introduz um novo método de aumento de dados para treinamento de modelos de ASR. O método proposto explora síntese de fala multilíngue e conversão de voz *cross-lingual*;

- O método de aumento de dados proposto alcança resultados comparáveis ao estado da arte do idioma Inglês em dois idiomas alvos utilizando apenas um locutor no treinamento do modelo. Deste modo, tornando mais viável a aplicação de síntese de fala na geração de datasets para o treinamento de modelos ASR em linguagens com poucos recursos disponíveis;

- Mostra que com o uso do método de aumento de dados proposto foi possível obter resultados promissores no treinamento de um modelo ASR utilizando apenas um único locutor real no idioma alvo.

A single speaker is almost all you need for automatic speech recognition

Edresson Casanova[1,2], *Christopher Shulby*[3], *Arnaldo Candido Junior*[4], *Sandra Aluísio*[1], *Moacir Antonelli Ponti*[1,5]

[1] Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Brazil
[2] Coqui, Germany
[3] Defined.ai., United States of America
[4] Federal University of Technology – Paraná, Brazil
[5] Mercado Livre, Brazil.

edresson@usp.br

Abstract

We explore cross-lingual multi-speaker speech synthesis and cross-lingual voice conversion applied to data augmentation for automatic speech recognition (ASR) systems. Through extensive experiments, we show that our approach permits the application of speech synthesis and voice conversion to improve ASR systems on a target language using only one target-language speaker during model training. We managed to close the gap between ASR models trained with synthesized versus human speech compared to other works that use many speakers. Finally, we show that it is possible to obtain promising ASR training results with our data augmentation method using only a single real speaker in a target language.

Index Terms: Speech Recognition, Speech Synthesis, Cross-lingual Zero-shot Voice Conversion, Cross-lingual Zero-shot Multi-speaker TTS, ASR Data Augmentation

1. Introduction

Text-to-Speech (TTS) systems have received a lot of attention in recent years due to the great advances in deep learning, which have allowed for the massive use of voice-based applications such as virtual assistants. These advances have allowed TTS models to achieve naturalness with respect to human speech [1, 2, 3]. Notably, most TTS systems are tailored for a single speaker, but many applications could benefit from the new-speaker synthesis, i.e., not seen during training, employing only a few seconds of the target speech. This approach is called zero-shot multi-speaker TTS (ZS-TTS) as in [4, 5, 6].

Advances in TTS have motivated works that exploit it as a way to improve Automatic Speech Recognition (ASR) systems. Researchers have explored two different approaches. The first is parallel training of ASR and TTS models; in this approach the TTS and ASR systems can improve themselves, as in [7, 8, 9]. The second is the use of a pre-trained TTS model to generate new data to train the ASR, such as [10], [11] and [12]. In this work, we will focus on the second approach.

All the works that explore a pre-trained TTS model to generate data for ASR explored the use of the LibriSpeech [13] dataset to train the ASR model. For the TTS model training, [10] used 3 speakers from the American English MAILABS [14] dataset, while [11] and [12] trained the TTS model with more than 251 speakers from the LibriSpeech.

In Table 1 we report the Word Error Rate (WER) of the best experiment of each of the related works in the test-other subset of the LibriSpeech dataset. Although the studies contain both test-clean and test-other results, we focus on the results of the most difficult sub-set. Also, [12] reported results using an external language model (LM); however, for fairness, we omit this LM in our comparison.

The ASR model trained with synthesized speech combined with human speech achieved relative improvement [1] of 4.56%, 0.79% and 4.25% than the model trained with human speech alone, respectively, for [10], [11] and [12]. Despite that, the relative difference between the model trained with only human speech and only synthesized speech was 80.17% and 78.98%, respectively, for [10] and [11], which motivates further improvements and research.

Although previous work shows the potential for multi-speaker or zero-shot multi-speaker TTS models for ASR data augmentation, these models still require high-quality datasets with many speakers and hours of speech to converge [12]. Generally, such models are trained on English with big datasets such as VCTK [20] and LibriTTS [21]. This approach is not suitable for low-resource languages. The vast majority of languages do not have an open TTS dataset and the few available are limited in the number of speakers.

In our previous work [6], we presented YourTTS a multilingual zero-shot multi-speaker TTS model that showed good results for Portuguese using only a single-speaker dataset in the target language.

In this paper, we combine the power of the YourTTS model with that of Wav2vec 2.0 [22], trained in a self-supervised way on 100 thousand hours of speech in 23 different languages [23], in order to show the viability of the application of TTS data augmentation for ASR systems in languages using just one speaker during TTS training. We also propose a new augmentation approach that uses cross-lingual voice conversion and cross-lingual multi-speaker TTS to alleviate the lack of speakers in a target language. We showed that with our approach, it is possible to train an ASR model using just a single-speaker, achieving promising results.

2. Audio datasets

We used 3 languages/training datasets for the TTS model:

English: VCTK [20] dataset, containing 44 hours of speech from 109 speakers, sampled at 48KHz. We divided the VCTK dataset into training, development and test subsets following [6]. To further increase the number of speakers for training,

[1] although relative improvement/difference is not widely used, the improvement achieved in related works is not easily comparable due to approaches' differences. For this reason, we used this metric, showing the real improvement achieved by related works approaches.

Table 1: *Comparison between related works in the test-other subset.*

Paper	TTS Model	ASR Model	Train data	WER
[10]	Tacotron GST [15]	Wav2Letter [16]	Human	16.21
			Synthesized	81.78
			Human + Synthesized	15.47
[11]	ZS-Tacotron [4] + VAE [17]	LAS [18]	Human	13.89
			Synthesized	66.10
			Human + Synthesized	13.78
[12]	GMVAE Tacotron [17]	LAS [18, 19]	Human	14.10
			Synthesized	–
			Human + Synthesized	13.50

we used the subsets *train-clean-100* and *train-clean-360* from LibriTTS [21]. In the end, our TTS model was trained with 1,248 English speakers.

Portuguese: TTS-Portuguese Corpus [24], a single-speaker male dataset in Brazilian Portuguese (pt-BR) containing ca. 10 hrs, sampled at 48KHz. As the authors did not use a soundproof studio, the dataset contains some environmental noise. Following [6], we resampled the audios to 16Khz and used FullSubNet [25] as a denoiser. For development, we randomly selected 500 samples, leaving the rest for training.

Russian: ru_RU set of the M-AILABS dataset [14], which is based on LibriVox[2]. The dataset consists of 46 hrs from 1 female and 2 male speakers. In this work, we used samples only from the female speaker for diversity, since we already used a male for pt-BR.

For all TTS datasets, pre-processing was carried out to normalize volume and to remove long silence periods, following [6]. After pre-processing, the datasets contained 8.38 hrs for pt-BR and 14.94 hrs for ru-RU (Russian).

For ASR model training, we used Common Voice version 7.0 [26] for pt-BR and ru-RU. In all experiments, we used the default train, development and test partitions. For pt-BR, these sets have 14.52, 8.9 and 9.5 hours, respectively; and ru-RU has 25.95, 13.06 and 13.75 hours, in the same order.

3. Audio augmentation methods

We explore three popular augmentation methods in speech processing – additive noise, pitch shifting and room impulse response (RIR) simulation. For additive noise, we use the MUSAN corpus [27], which is composed of three subsets: (i) the noise subset (ca. 6 hrs.), such as ambient noise and dialtones; (ii) the music subset (ca. 42 hrs.); and (iii) the speech subset (ca. 60 hrs). For RIR, we use the simulated filters provided in [28].

As in [29], for additive noise, we randomly selected the audio and added it to the original signal. For the speech subset, we added it to the original signal with a random signal to noise ratio (SNR) from 13 to 20dB. In a similar fashion, we added from 5 to 15dB SNR for music and from 0 to 15dB SNR for noise. For the RIR filters, we performed speech reverberation via convolution operation with a collection of simulated RIR filters [28]. For pitch shift, we randomly chose a semitone from -4 to 4. All augmentations are randomly selected with a 25% probability of being chosen for each audio in every training step. For all methods, we use the implementations available in the Python Audiomentations[3] package.

4. TTS model setup

We use YourTTS, [6] fine-tuned in English, pt-BR and ru-RU. In the original work, YourTTS was trained using English (LibriTTS and VCTK corpora), French (M-AILABS corpus) and pt-BR (TTS-Portuguese Corpus). The original model was trained using only one male speaker in pt-BR but still produced good results in zero-shot multi-speaker TTS and zero-shot voice conversion for pt-BR. Furthermore, it was able to produce female voices even without being trained on female voices, making it adequate for the objective of this study.

In this work, we use transfer learning from the checkpoints made publicly available by the authors. The dataset in English and pt-BR were the same dataset used in the YourTTS [6] original paper. One difference should be noted: we replaced the French M-AILABS dataset with a female speaker from the Russian M-AILABS dataset to attend the experiment requirements as explained in Section 5.1. Using these datasets, we train the YourTTS model for 100k steps with the same parameters used in the original work. After this, we fine-tuned the model for another 40k steps using the Speaker Consistency Loss (SCL) [6]. Therefore, in this work YourTTS was trained with 1,248 speakers in English, one male speaker in Portuguese and one female speaker in Russian.

YourTTS can synthesize different audios for the same input sentence. During inference, the latent variable predicted by the text encoder is added with a random sample of the standard normal distribution multiplied by a temperature T. In this way, diversity can be controlled by the temperature T. As shown by [30], the manipulation of T allows generating different pitches; for more details see [30, 3, 6]. Furthermore, the YourTTS model is trained with the stochastic flow-based duration predictor proposed in [3]. As shown in [3], this duration predictor can produce several different speech rhythms for the same sentence. This happens because during inference, a random sample of the standard normal distribution is multiplied by a temperature T_{dp} and added to the latent variable before it is inverted by the flow. In this way, it is possible to control the variety of rhythms with the temperature T_{dp} [3]. Finally, it is possible to control the speaking rates by multiplying the predicted durations by a positive scalar L, thus making the pronunciation faster when L is smaller and slower when L is bigger. During the generation of

[2] https://librivox.org/

[3] https://github.com/iver56/audiomentations

the artificial datasets, diversity is achieved by randomly choosing L, T and T_{dp}: for L, a value between 0.7 and 2, while for temperatures (T and T_{dp}) a value between 0 and 0.667.

5. Comparison between human and synthesized speech

Previous works have shown a large gap between ASR models trained with human and synthesized speech [11, 12]. In these works, the authors have used a large number of speakers and hours of speech in the target language during the TTS model training. On the other hand, to apply this method here for languages with low/medium resources, we proposed the use of only a single speaker in the target languages during the TTS model training, in order to verify whether being trained even with only a single speaker in the target language, the YourTTS model can be used as the data augmentation for ASR, in this section we compared ASR models trained with synthesized speech and human speech.

5.1. Experiments Setup

For a fair comparison between human and synthesized speech, we have generated a copy of pt-BR and ru-RU Common Voice corpora. For each sentence in Common Voice, we generate the pronunciation of that sentence in the voice of the same speaker, using the pronunciation of that sentence as a reference to extract the speaker embedding. The idea being that if the zero-shot multi-speaker TTS model is good enough, it will generate the same speaker's voice as in the original audio.

As for the ASR model, we use Wav2vec 2.0 [22], a large model trained in a self-supervised way on the VoxPopuli dataset [23]. We used the model checkpoint provided by the authors at [23] which was trained on 100,000 hours of speech in the following 23 languages: Bulgarian (Bg), Czech (Cs), Croatian (Hr), Danish (Da), Dutch (Nl), English (En), Estonian (Et), Finnish (Fi), French (Fr), German (De), Greek (El), Hungarian (Hu), Italian (It), Latvian (Lv), Lithuanian (Lt), Maltese (Mt), Polish (Pl), Portuguese (Pt), Romanian (Ro), Slovak (Sk), Slovene (Sl), Spanish (Es) and Swedish (Sv). We chose Portuguese and Russian because the Portuguese language was used in the pre-training of this model and Russian was not used, presenting realistic results for both scenarios. Also, these languages are from diverse language families. We carried out four experiments:

- **Experiment 1:** ASR models trained in pt-BR and ru-RU with the Common Voice dataset using the standard training and development subsets. For pt-BR, the model was trained for 140 epochs and ru-RU for 100 epochs;
- **Experiment 1.1:** Transfer learning from experiment 1 and adds in the training audio augmentations (AA) described in Section 3. In this experiment, the ASR is trained with half the number of epochs used in experiment 1.
- **Experiment 2:** Similar to experiment 1, but the model is trained using the synthesized copy of Common Voice in pt-BR and ru-RU. For model training and development, we used synthesized speech.
- **Experiment 2.1:** Transfer learning from experiment 2 and adds the AA described in Section 3. The ASR is trained with half the number of epochs used in experiment 2.

To run the experiments, we use the HuggingFace Transformers framework[4]. The models were trained with a GPU NVIDIA TITAN RTX 24GB using a batch size of 8 and gradient accumulation over 24 steps. We used the optimizer AdamW [31] with a linear learning rate warm-up from 0 to 3e-05 in the first 8 epochs and after using linear decay to zero. During training, the best checkpoint was chosen, using the loss in the development set and used early stopping after the development loss had not improved for 10 consecutive epochs. The code used for all of the experiments, as well as the checkpoints of the trained models are publicly available at: `https://github.com/Edresson/Wav2Vec-Wrapper`.

5.2. Results and Discussion

Table 2 presents the results for our experiments on the pt-BR and ru-RU test subsets from Common Voice.

Table 2: *WER of the comparison between human and synthesized speech*

Exp.	PT	RU
1. Human Speech	23.50	25.47
1.1 Human Speech + AA	21.54	22.27
2. Synthesized speech	56.84	65.85
2.1 Synthesized speech + AA	43.99	50.46

The model trained only with human speech (Experiment 1) reached a WER of 23.50% and 25.47%, respectively for pt-BR and ru-RU. The model trained only with synthesized speech (Experiment 2) reached a WER of 56.84% and 65.85%, respectively, for pt-BR and ru-RU. Therefore, without AA, the relative difference between the model trained with only human speech and only synthesized speech is of 58.65% and 61.32% for those two languages.

As expected, fine-tuning the models with AA (Experiment 1.1 and Experiment 2.1) improved performance. The model trained with human speech only improved its result by 1.96% and 3.20% WER for pt-BR and ru-RU after the addition of AA. The model trained only with synthesized speech improved by 12.85% and 15.39% WER, respectively. Therefore, using AA benefits the model trained with synthesized speech much more than the model trained with human speech. This can be explained by the absence of noise diversity in the synthesized speech. Common Voice is a dataset composed of a lot of environmental noises, whereas the synthesized speech just has some artifacts, but not environmental noises. Therefore, with the use of AA, the gap between models trained only with human and only synthesized speech is reduced to a relative difference of 51.03% and 55.86% for those two languages.

Our results are interesting because, despite using only a single speaker dataset for training the TTS model for pt-BR and ru-RU, our ASR model trained only with synthesized speech achieves a comparable result to related works. For comparison, considering the results reported by [11], the relative difference between models trained only with human and only synthesized speech was of 78.98%. Even though this work explores the English language in a setting with many available speakers and it is not directly comparable, we believe that our results indicate that our approach requiring just one speaker in the target language, can be used for low-resource languages.

[4] https://github.com/huggingface/transformers

Table 3: *WER of the experiments on the test subsets of the pt-BR and ru-RU Common Voice datasets*

Experiment	Train data	PT	RU
Baseline	TTS dataset (single-speaker)	63.90	74.02
Upper Bound	Common Voice + TTS dataset	20.39	24.80
Baseline + DA	TTS dataset + *GEN_TTS* + *GEN_VC*	33.96	36.59

6. Is only one speaker in the target language sufficient for learning?

In section 5 we applied the YourTTS model, trained with a single speaker in pt-BR and ru-RU, to create datasets for ASR. Although the TTS model was trained with only one speaker, we made a copy of the Common Voice dataset in order to gain many in the target languages. This approach has shown promising results, but many languages do not have datasets with many available speakers. For that reason, we explore the use of a single-speaker in the target language, for training both TTS and ASR. To make up for the lack of speakers during the creation of the synthesized dataset, we explored the use of voice conversion, using English speakers, rather than cloning speakers from the target languages.

6.1. Experiments Setup

We created the *GEN_TTS* dataset by synthesizing all the sentences in Common Voice using English speakers. Furthermore, we created the *GEN_VC* dataset which consists of the single-speaker used for training the TTS model in the target language converted to a multi-speaker dataset using cross-lingual voice conversion with English speakers. For voice conversion, we also use the YourTTS model. Each sample of the dataset used in the TTS model training was converted to the voice of 5 speakers, chosen at random from the 1,248 English speakers used to train the YourTTS model. The value of 5 transfers per sample was chosen in preliminary experiments. We would like to note that, in preliminary experiments, we explored increasing the number of speakers per sentence in the *GEN_TTS* dataset; however, this did not bring significant improvements. We carried out three experiments:

- **Baseline:** ASR models trained with the single-speaker dataset used during the TTS model training on pt-BR and ru-RU.
- **Upper Bound:** ASR models trained on pt-BR and ru-RU with Common Voice plus the single-speaker TTS dataset.
- **Baseline + DA:** explores the use of human speech from a single-speaker in the target language (TTS dataset), with data augmentation (DA) being accomplished by the YourTTS model. For the data augmentation we merge the *GEN_TTS* and *GEN_VC* datasets, detailed above.

All 3 experiments use AA and the same training parameters used in Section 5.

6.2. Results and Discussion

Table 3 presents our experiments' results on the test subsets of the pt-BR and ru-RU Common Voice datasets.

The model trained with just a single-speaker in the target languages (Baseline) achieved a WER of 63.90% and 74.02% for pt-BR and ru-RU, respectively. The model trained with only 1 real speaker in the target language (TTS dataset) with data augmentation using voice conversion and speech synthesis (Baseline + DA), achieved a WER of 33.96% and 36.59%, respectively. Therefore, our data augmentation approach in scenarios with just one real speaker available improves the WER by 29.94% and 37.43% for pt-BR and ru-RU, respectively.

Comparing the results with the SOTA English ASR system on the Common Voice dataset (7.7% achieved by [32]) these results do not look so remarkable. However, for comparison in pt-BR, [33] used ca. 158 hrs of speech and a non-self-supervised model without an external LM and achieved a WER of 47.41% on the test set of BRSD v2 dataset. Despite being different datasets, [34] showed that the Common Voice test set is more challenging than the test set of BRSD v2, and for this reason, the model proposed by these authors reached a higher WER on the Common Voice dataset. In ru-RU, [35] used transfer learning from 5 large English datasets, they trained the QuartzNet [36] model on Common Voice, obtaining a WER of 32.20% on the test set of the same dataset. Therefore, 33.96% WER achieved by our model is probably superior to the SOTA for pt-BR, before the introduction of self-supervised learning approaches. Also, the WER of 36.59% achieved in Russian is comparable with the SOTA.

Comparing the results of the Baseline + DA experiment with the Upper Bound (20.39% and 24.80% for pt-BR and ru-RU). Our results are still a little far from the Upper Bound. However, the results are remarkable since our model was trained with just **one real speaker in the target language**.

7. Conclusions and future work

In this work, we present a novel data augmentation approach for ASR training by using cross-lingual speech synthesis and voice conversion. We show that it is possible to achieve promising results for ASR model training with just a single-speaker dataset in a target language, making it viable for a low-resource scenario. Finally, our approach works both in a language (pt-BR) present in the Wav2Vec 2.0 model pre-training, as well as for an unseen language (Russian).

In future work, we intend to explore the use of the self-supervised model feature extractor as a discriminator during the training of the YourTTS model. In this way, the YourTTS model may produce even more human-like speech in the self-supervised model.

8. Acknowledgements

This study was funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Finance Code 001, by CNPq (National Council of Technological and Scientific Development) grant 304266/2020-5, by Artificial Intelligence Excellence Center (CEIA)[5]. The authors of this work also would like to thank the Center for Artificial Intelligence (C4AI-USP)[6] and the support from the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and from the IBM Corporation.

[5] http://centrodeia.org

[6] https://c4ai.inova.usp.br/

9. References

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

[2] R. Valle, K. J. Shih, R. Prenger, and B. Catanzaro, "Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis," in *International Conference on Learning Representations*, 2020.

[3] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5530–5540.

[4] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.

[5] S. Choi, S. Han, D. Kim, and S. Ha, "Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding," *arXiv preprint arXiv:2005.08484*, 2020.

[6] E. Casanova, J. Weber, C. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone," *arXiv preprint arXiv:2112.02418*, 2021.

[7] A. Tjandra, S. Sakti, and S. Nakamura, "Machine speech chain with one-shot speaker adaptation," *Proc. Interspeech 2018*, pp. 887–891, 2018.

[8] ——, "Listening while speaking: Speech chain by deep learning," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 301–308.

[9] ——, "Machine speech chain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 976–989, 2020.

[10] J. Li, R. Gadde, B. Ginsburg, and V. Lavrukhin, "Training neural speech recognition systems with synthetic speech augmentation," *arXiv preprint arXiv:1811.00707*, 2018.

[11] A. Rosenberg, Y. Zhang, B. Ramabhadran, Y. Jia, P. Moreno, Y. Wu, and Z. Wu, "Speech recognition with augmented synthesized speech," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 996–1002.

[12] A. Laptev, R. Korostik, A. Svischev, A. Andrusenko, I. Medennikov, and S. Rybin, "You do not need more data: improving end-to-end speech recognition by text-to-speech data augmentation," in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, 2020, pp. 439–444.

[13] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[14] Munich Artificial Intelligence Laboratories GmbH, "The m-ailabs speech dataset – caito," 2017. [Online]. Available: https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/

[15] Y. Wang, D. Stanton, Y. Zhang, R.-S. Ryan, E. Battenberg, J. Shor, Y. Xiao, Y. Jia, F. Ren, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," in *International Conference on Machine Learning*. PMLR, 2018, pp. 5180–5189.

[16] R. Collobert, C. Puhrsch, and G. Synnaeve, "Wav2letter: an end-to-end convnet-based speech recognition system," *arXiv preprint arXiv:1609.03193*, 2016.

[17] W.-N. Hsu, Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Y. Wang, Y. Cao, Y. Jia, Z. Chen, J. Shen *et al.*, "Hierarchical generative modeling for controllable speech synthesis," *arXiv preprint arXiv:1810.07217*, 2018.

[18] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4960–4964.

[19] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N.-E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," *Proc. Interspeech 2018*, pp. 2207–2211, 2018.

[20] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2016.

[21] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.

[22] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[23] C. Wang, M. Riviere, A. Lee, A. Wu, C. Talnikar, D. Haziza, M. Williamson, J. Pino, and E. Dupoux, "VoxPopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 993–1003. [Online]. Available: https://aclanthology.org/2021.acl-long.80

[24] E. Casanova, A. C. Junior, C. Shulby, F. S. d. Oliveira, J. P. Teixeira, M. A. Ponti, and S. Aluísio, "Tts-portuguese corpus: a corpus for speech synthesis in brazilian portuguese," *Language Resources and Evaluation*, pp. 1–13, 2022.

[25] X. Hao, X. Su, R. Horaud, and X. Li, "Fullsubnet: A full-band and sub-band fusion model for real-time single-channel speech enhancement," *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Jun 2021. [Online]. Available: http://dx.doi.org/10.1109/ICASSP39728.2021.9414177

[26] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the 12th Language Resources and Evaluation Conference*, 2020, pp. 4218–4222.

[27] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[28] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5220–5224.

[29] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova baseline system for the voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2009.14153*, 2020.

[30] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *arXiv preprint arXiv:2005.11129*, 2020.

[31] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017.

[32] Y. Zhang, D. S. Park, W. Han, J. Qin, A. Gulati, J. Shor, A. Jansen, Y. Xu, Y. Huang, S. Wang *et al.*, "Bigssl: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[33] I. M. Quintanilha, S. L. Netto, and L. W. P. Biscainho, "An open-source end-to-end asr system for brazilian portuguese using dnns built from newly assembled corpora," *Journal of Communication and Information Systems*, vol. 35, no. 1, pp. 230–242, 2020.

[34] L. R. Stefanel Gris, E. Casanova, F. Santos de Oliveira, A. da Silva Soares, and A. C. Junior, “Brazilian portuguese speech recognition using wav2vec 2.0,” *arXiv e-prints*, pp. arXiv–2107, 2021.

[35] J. Huang, O. Kuchaiev, P. O’Neill, V. Lavrukhin, J. Li, A. Flores, G. Kucsko, and B. Ginsburg, “Cross-language transfer learning, continuous learning, and domain adaptation for end-to-end automatic speech recognition,” *arXiv preprint arXiv:2005.04290*, 2020.

[36] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, “Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions,” in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6124–6128.

A. Upper Bound + DA

To verify how data augmentation using speech synthesis and voice conversion can improve the results of ASR models trained with multiples humans speakers, even when the TTS/voice conversion model was trained with only a single speaker in the target language. We train the ASR models with the merged datasets from *GEN_TTS*, *GEN_VC*, TTS dataset and Common Voice. We called this experiment as Upper Bound + DA.

Table 4 presents results of the experiments Upper Bound (reported previously on Section 6) and Upper Bound + DA experiments on the test subsets of the pt-BR and ru-RU Common Voice datasets.

Table 4: *WER of Upper Bound and Upper Bound + DA experiments on the test subsets of the pt-BR and ru-RU Common Voice datasets*

Experiment	**Train data**	**PT**	**RU**
Upper Bound	Common Voice + TTS dataset	20.39	24.80
Upper Bound + DA	Common Voice + TTS dataset + GEN_TTS + GEN_VC	20.20	19.46

The model trained with Common Voice and a single-speaker TTS dataset in the target languages (Upper Bound) achieved a WER of 20.39% and 24.80% for pt-BR and ru-RU, respectively. The model trained with Common Voice, a single-speaker TTS dataset and our data augmentation approach in the target languages (Upper Bound + DA) achieved a WER of 20.20% and 19.46%, respectively. Therefore, the ASR model's trained with our data augmentation approach achieved a relative improvement of 0.93% and 21.53%, respectively, for those languages. The relative improvement achieved in pt-BR is consistent with the results reported in [11], where a relative improvement was of 0.79%. However, it is lower than the result reported by [10] (4.56%) and [12] (4.25%). On other hand, the relative improvement achieved in ru-RU is significantly superior than taht which was reported in related works.

Unlike related works, we use AA and only one target-language speaker in the training of the TTS/voice conversion model. AA can make the ASR model more robust and improve generalization; however, these approaches may have overlapping contributions. To verify this hypothesis, we did an ablation study by re-training the Upper Bound and Upper Bound + DA experiments in pt-BR without AA. The Upper Bound achieved a WER of 22.96% and the Upper Bound + DA achieved a WER of 21.41%. That is, without the use of AA as in the related works, the relative difference between ASR models trained only with human and only synthesized speech in pt-BR is 6.75%. Thus, in pt-BR and ru-RU, our approach achieves results better than related works in English, using only one speaker for training the TTS/voice conversion model in the target language. In this way, we have shown that it is possible to apply TTS systems to ASR dataset generation even for languages where just a single-speaker dataset is available.

We believe that the difference between the results achieved for pt-BR and ru-RU, can be explained by the characteristics of the datasets. In Common Voice, the amount of hours available for training the ASR model for the ru-RU is 25.95 hrs as opposed to 14.52 hrs for pt-BR. Furthermore, the ru-RU TTS dataset has approximately 6.5 more hours after preprocessing than pt-BR and the ru-RU TTS dataset is high-quality. In our experiments, we use voice conversion to transform the TTS dataset into a multi-speaker dataset using 5 transfers for each sample, thus the difference in the number of hours is a multiple as well.

CAPÍTULO

8

CORAA: A LARGE CORPUS OF SPONTANEOUS AND PREPARED SPEECH MANUALLY VALIDATED FOR SPEECH RECOGNITION IN BRAZILIAN PORTUGUESE

Título:	**CORAA: a large corpus of spontaneous and prepared speech manually validated for speech recognition in Brazilian Portuguese**
Autores:	**Arnaldo Candido Junior, Edresson Casanova, Anderson Soares, Frederico Santos de Oliveira, Lucas Oliveira, Ricardo Corso Fernandes Junior, Daniel Peixoto Pinto da Silva, Fernando Gorgulho Fayet, Bruno Baldissera Carlotto, Lucas Rafael Stefanel Gris, Sandra Maria Aluísio**
Ano:	**Submetido em outubro de 2021; Aceito com revisões em fevereiro de 2022**
Revista:	**Language Resources and Evaluation (LREV)**
Situação:	**Aceito com revisões; a versão revisada (diferente da versão abaixo) foi submetida em abril de 2022**

Motivação:

Recentemente grandes *datasets* foram disponibilizados publicamente para o treinamento de modelos de ASR no idioma Português Brasileiro (ARDILA *et al.*, 2019; QUINTANILHA; NETTO; BISCAINHO, 2020; ELIZABETH *et al.*, 2021). Deste modo, o Português Brasileiro passou a ter aproximadamente 574 horas de dados de fala que podem ser usados para treinar novos modelos ASR. Apesar disso, ainda faltam *datasets* com gravações de fala espontânea de vários gêneros, como por exemplo entrevistas, diálogos e conversas informais, ou seja, conversas gravadas em contextos naturais e ambientes ruidosos. A fala espontânea apresenta vários fenômenos como risos, tosses, pausas preenchidas, fragmentos de palavras resultantes de repetições, recomeços e revisões do discurso. A falta de *datasets* compostos por fala espontânea dificulta o desenvolvimento de sistemas de reconhecimento automático de fala capazes de lidar com fala espontânea gravada em ambientes ruidosos. Por esse motivo, nesse trabalho foi construído e disponibilizado publicamente um *dataset* contendo 290,77 horas de fala espontânea e preparada.

Contribuições relevantes do artigo:

- Apresenta um grande dataset para a tarefa de ASR no Português Brasileiro, contendo 290,77 horas de fala espontânea e preparada. O dataset também inclui 4,69 horas de fala preparada no Português Europeu;
- Disponibiliza publicamente um modelo ASR treinado no dataset proposto.

CORAA: a large corpus of spontaneous and prepared speech manually validated for speech recognition in Brazilian Portuguese

Arnaldo Candido Junior[1], *Edresson Casanova*[2], *Anderson Soares*[3], *Frederico Santos de Oliveira*[3], *Lucas Oliveira*[1], *Ricardo Corso Fernandes Junior*[1], *Daniel Peixoto Pinto da Silva*[1], *Fernando Gorgulho Fayet*[2], *Bruno Baldissera Carlotto*[2], *Lucas Rafael Stefanel Gris*[1], *Sandra Maria Aluísio*[2]

[1] Federal University of Technology – Paraná, Brazil
[2] Instituto de Ciências Matemáticas e de Computação, University of São Paulo, Brazil
[3] Federal University of Goias, Brazil
arnaldocan at gmail dot com

Abstract

Automatic Speech recognition (ASR) is a complex and challenging task. In recent years, there have been significant advances in the area. In particular, for the Brazilian Portuguese (BP) language, there were about 376 hours public available for ASR task until the second half of 2020. With the release of new datasets in early 2021, this number increased to 574 hours. The existing resources, however, are composed of audios containing only read and prepared speech. There is a lack of datasets including spontaneous speech, which are essential in different ASR applications. This paper presents CORAA (Corpus of Annotated Audios) v1. with 290.77 hours, a publicly available dataset for ASR in BP containing validated pairs (audio-transcription). CORAA also contains European Portuguese audios (4.69 hours). We also present a public ASR model based on Wav2Vec 2.0 XLSR-53 and fine-tuned over CORAA. Our model achieved a Word Error Rate of 24.18% on CORAA test set and 20.08% on Common Voice test set. When measuring the Character Error Rate, we obtained 11.02% and 6.34% for CORAA and Common Voice, respectively. CORAA corpora were assembled to both improve ASR models in BP with phenomena from spontaneous speech and motivate young researchers to start their studies on ASR for Portuguese. All the corpora are publicly available at `https://github.com/nilc-nlp/CORAA` under the CC BY-NC-ND 4.0 license.

Index Terms: Automatic Speech Recognition, Spontaneous Speech, Prepared speech, Brazilian Portuguese, Public Datasets, Public Speech Corpora

1. Introduction

Automatic Speech Recognition (ASR) is a complex and challenging. Significant progress in techniques, models for the task had occurred in recent years. The main reasons for this progress include (but are not limited to) the availability of large-scale datasets and advances in deep learning methods running over powerful computing platforms.

Despite significant advances in ASR benchmarking solutions, the main and large datasets available for training and evaluating ASR systems are English due to the predominance of the language in science and business, although there are some current efforts to build multilingual speech corpora [1, 2, 3, 4]. Another problem is the environment of the recording, mostly composed of clean speech. Regarding the style of speaking, they are read speech, such as [5, 4, 1, 2, 6] or prepared speech like [7, 8].

In this paper, we focus on a specific language – the Brazilian Portuguese (BP) –, which was struggling with only a few dozen hours of public data available until the middle of 2020. The previous open dataset to train speech models in BP were much smaller than American English datasets, with only 10 hours for speech synthesis (TTS)[1] and 60 hours for ASR. The resource commonly used to train ASR models for BP is an ensemble of four small, non-conversational speech datasets: the Common Voice Corpus version 5.1 (Mozilla)[2], Sid dataset[3], VoxForge[4], and LapsBM1.4[5].

In the second half of 2020, three new datasets were made available: (i) the BRSD v2 which includes the CETUC dataset [9] (with almost 145 hours), plus 12 hours and 30 minutes of non-conversational speech from 3 small open datasets[6] [10], (ii) the Multilingual LibriSpeech (MLS), derived from reading LibriVox audiobooks in 8 languages, including BP [4] with 169 hours, and (iii) the dataset Common Voice version 6.1[7] [1], with 50 validated hours, composed of recordings of read sentences which were displayed on the screen. These three datasets total 376 hours. Given this recent public availability of large audio databases for BP language, the lack of resources has been gradually reduced, although it is still far from ideal when compared to resources for the English language.

In early 2021, a new dataset with prepared speech, called the Multilingual TEDx Corpus [8], was made publicly available, providing 765 hours to support speech recognition and speech translation research. The Multilingual TEDx Corpus is composed by a collection of audio recordings from TEDx talks in 8 source languages, including 164 hours of Portuguese. Moreover, a new version of the dataset Common Voice (Common Voice Corpus 7.0) was launched with 84 validated hours, which is an increment of 34 hours over the previous version. Therefore, currently, BP language is well represented with 574 hours of speech data which can be used to train new ASR models.

However, there is still a lack of datasets with audio files that record spontaneous speech of various genres, from interviews

[1] `https://github.com/Edresson/TTS-Portuguese-Corpus`

[2] `https://commonvoice.mozilla.org/pt/datasets`

[3] `https://doi.org/10.17771/PUCRio.acad.8372`

[4] `http://www.voxforge.org/pt/downloads`

[5] `https://laps.ufpa.brfalabrasil/`

[6] BRSD v2 also includes CSLU: Spoltech Brazilian Portuguese Version 1.0 — `https://catalog.ldc.upenn.edu/LDC2006S16`

[7] `https://commonvoice.mozilla.org/pt/datasets`

to informal dialogues and conversations, i.e., conversational speech recorded in natural contexts and noisy environments to train robust ASR systems. Spontaneous speech presents several phenomena such as laughter, coughs, filled pauses, word fragments resulted from repetitions, restarts and revisions of the discourse. This gap makes difficult the development of both high-quality dialog systems and automatic speech recognition systems capable of handling spontaneous speech recorded in noisy environments. The latter ones are called rich transcription-style ASR (RT-ASR) when they explicitly convert those phenomena cited above into special tokens [11, 12, 13]. Dialog systems, for example, must deal with several types of speech disfluencies, preserving them instead of removing filled pauses and word fragments [14]. In general, it is expected that ASR systems trained on read style and clean speech will face a drop of performance when dealing with informal conversations in contexts of free interactions and noisy environments.

The TaRSila project is an effort of the Center for Artificial Intelligence[8] (C4AI) to make available language resources to bring natural language processing of BP to the state-of-the-art. The project aims at growing speech datasets for BP language, to achieve state-of-the-art results for automatic speech recognition, multi-speaker synthesis, speaker identification, and voice cloning. In a joint effort of two research centers, the C4AI and the CEIA[9] (Center of Excellence in Artificial Intelligence, in English), four speech corpora composed of **prepared, guided interviews and spontaneous speech** from academic projects were manually validated to serve as an ASR benchmark for BP. The projects are: (i) ALIP [15]; (ii) C-ORAL Brasil I [16]; (iii) Nurc-Recife [17]; and (iv) SP2010 [18]. We also validated 76.36 hours of prepared speech from a collection of TEDx Talks[10] in Brazilian Portuguese, including 4.69 hours of European Portuguese, to allow experiments with Portuguese language variants.

1.1. Goals

In this paper we present a new public available dataset called CORAA (Corpus of Annotated Audios) v1. CORAA has 290.77 hours of validated pairs (audio-transcription) and is composed by five corpora: ALIP [15], C-ORAL Brasil I [16], NURC-Recife [17], SP2010 [18], TEDx Portuguese talks. Information about each corpora is presented in Table 1. The original sizes of each dataset in hours are presented as reported in their respective original papers, when reported by the authors. Regarding SP2010, the total duration is estimated, since the authors report 60 recordings from 60 to 70 minutes each and the total hours of ALIP was computed after download. All the corpora are publicly available[11] at `https://github.com/nilc-nlp/CORAA` under the CC BY-NC-ND 4.0 license. These corpora were assembled with the purpose of improving ASR models in BP with phenomena from spontaneous speech and noise in order to motivate young researchers in this exciting research area.

As an example of the feasibility of speech recognition with CORAA, we present one speech recognition experiment using the Wav2vec 2.0 XLSR-53 [19, 20]. Furthermore, we compared our model with the state of the art in automatic speech recognition in Brazilian Portuguese[21]. This two models are evaluated according to three main scenarios: (a) testing audios with different characteristics from training; (b) focusing on model performance for each of the five corpora, considering noise level and accent; (c) analyzing spontaneous and prepared speech styles impacts on the trained models.

1.2. Highligths

The main contributions made in this work are summarised as follows.

1. A large BP corpus of validated pairs (audio-transcription) containing 290.77 hours, composed of five corpora (ALIP, C-ORAL Brasil I, NURC Recife, SP2010, and TEDx Portuguese talks), adapted for the task of ASR in BP. We also include 4.69 hours of European Portuguese (in TEDx Portuguese).
2. The first corpus, according to our knowledge, tackling spontaneous speech for ASR in BP.
3. An ASR Model, publicly available, using the presented corpus.

Section 2 details both related work on datasets available for ASR in BP and the five spoken corpora projects used in CORAA v1. Section 3 describes the steps followed in preparing the CORAA corpus. Section 4 presents the creation of train, development and test splits of CORAA, the experiment on ASR for BP and an error analysis of our model. Finally, Section 5 presents the final remarks of the work.

2. Related Works on Speech Datasets and Spoken Corpora for BP

2.1. Open Datasets for Speech Recognition in BP

Three new datasets were released for BP at the end of 2020. CETUC dataset [9] contains 145 hours of 100 speakers, half males, and half females. The sentence set is composed of 1,000 sentences (3,528 words). The sentences are phonetically balanced and extracted from CETEN-Folha[12] corpus. Each speaker uttered all sentences from the sentence set exactly once. CETUC was recorded in a controlled environment, using a sample rate of 16kHz. The audios are publicly available[13], without an explicit license. Regarding the environment of recording and speaking style, CETUC delivers clean and read speech.

Common Voice Corpus 6.1, version pt_63h_2020-12-11, contains 63 hours of audio, 50 of which were considered validated. The dataset comprises 1,120 BP speakers, 81% males and 3% females (some audios are not sex labeled). The audios were collected using the Common Voice website[14] or using a mobile APP. The speakers read aloud sentences presented on the screen. A maximum of 3 contributors analyzed each pair audio-transcription, and simple voting is applied: two votes for acceptance validate the audio; two votes for rejection invalidate the audio. A given release may also contain samples that were analyzed but did not receive enough votes to be validated/invalidated — these samples have the status "OTHER" [1]. Releases are distributed under the CC-0[15] license and contain MP3 files, originally collected at 48kHz sampling rate but downsampled to 16kHz. The following metadata is also

[8] `http://c4ai.inova.usp.br/pt/nlp2-pt/`

[9] `http://centrodeia.org/`

[10] `https://www.ted.com/`

[11] Currently, only the test set is not available, because it will be released after an ASR Challenge involving CORAA

[12] `https://www.linguateca.pt/cetenfolha/index_info.html`

[13] `https://igormq.github.io/datasets/`

[14] `https://commonvoice.mozilla.org/pt/speak`

[15] `https://commonvoice.mozilla.org/pt/datasets`

Table 1: *Speech Genres, Accents, Speaking Styles and Hours (in decimal) in each original CORAA Corpora*

Corpus	**ALIP**	**C-ORAL Brasil I**	**NURC Recife**	**SP2010**	**TEDx Portuguese**
Speech Genres	Interviews, Dialogues	Monologues, Dialogues, Conversations	Dialogues, Interviews, Conference and Class Talks	Conversations, Interviews, Reading	Stage Talks
Speaking Styles	Spontaneous Speech	Spont. Speech	Spont. and Prepared Speech	Spont. and Read Speech	Prepared Speech
Accent	São Paulo State Cities	Minas Gerais	Recife	São Paulo Capital	Misc.
Original (hrs)	78	21.13	279	65	249

available: ID_speaker, path_audio_file, read_sentence, up_votes, down_votes, age, gender, accent. Where up_votes e down_votes refers to the voting result, and the last three fields are optional. Regarding the speaking style, Common Voice Corpus has read speech. As for recording environment, both noise level and sound clarity is very heterogeneous. The current version of the dataset (Common Voice Corpus 7.0) has 84 validated hours, 34 hours more than version 6.1.

The Multilingual LibriSpeech (MLS) dataset [4] is composed by audios extracted from Librivox[16] audiobooks. The Librivox project releases audiobooks in the public domain. MLS dataset encompasses eight languages, including BP, and is released under the CC BY 4.0[17] license. MLS can be used for developing both ASR and TTS models. There are 160.96 hours for training models, 3.64 hours for tuning and 3.74 for testing for Portuguese. It provides 26 male and 16 female speakers in the training dataset; 5 female, and 5 male speakers for tuning; and the same for testing. The audios were downsampled from 48kHz to 16kHz for easy processing. Regarding the environment of the recording and speaking style, MLS is made of clean and read speech.

In early 2021, a new dataset was made publicly available — the Multilingual TEDx Corpus, licensed under the CC BY-NC-ND 4.0[18]. This dataset has recordings of TEDx talks in 8 languages, BP being one of them, represented with 164 hours and 93K sentences. Each TEDx talk is stored as a 44 or 48kHz sampled wav file. Available metadata include source language, talk title, speaker name, audio length, keywords, and a short talk description. Multilingual TEDx Corpus was built to advance ASR and speech translation research, with multilingual models and baseline models being distributed for ASR and speech translation. Regarding the speaking style and the environment of the recording, Multilingual TEDx Corpus is composed of prepared and clean speech.

2.2. Spoken corpora projects used in CORAA

2.2.1. ALIP

The project ALIP[19] (Amostra Linguística do Interior Paulista – Language Sample of the Interior of São Paulo, in English) [15] was proposed in 2002, and coordinated by Prof. Sebastião Carlos Leite Gonçalves, from UNESP São José do Rio Preto. This project was responsible for building the database called Iboruna [22], composed of two types of speech samples:

- A sample of 151 interviews (each with about 20 minutes, being 76 male and 76 female voices) from the northwest region of the São Paulo state;
- Another sample consisting of 11 dialogues, involving from two to five informants. It was secretly recorded in contexts of free social interactions. This sample has 28 informants (10 men and 18 women).

This corpus totals 78 hours and it is characterized by the spontaneous speech of the linguistic variety of Brazilian Portuguese spoken in the interior of São Paulo. It was compiled between the years of 2004 and 2005. The informants, residents of 7 different cities, range in age from 7 to over 55 years, with a considerable variety of income and education.

The speech samples were recorded with GamaPower and PowerPack digital recorders. For interviews, the consent of the informants was obtained before recording, while, for the dialogues, dialogues, the consent was obtained after recording. The interviewer conducted the interviews, and the dialogues were free, with topics defined by the participant interactions.

The corpus is available for academic use without a defined license, but with defined Terms of Use and Privacy Policy[20]. It is available via download from the project website. The two types of samples have a dedicated folder for each, in the following formats. Each folder contains .mp3 files (the audios are sampled in 8kHz), as well as .doc and .pdf files (transcriptions, informant's socio-demographic information, among others). It is important to note that audio files are not aligned with their transcriptions.

2.2.2. C-ORAL Brasil I

C-ORAL Brasil I is a corpus published in 2012, resulting of the project C-ORAL Brasil[21] under the coordination of Tommaso Raso and Heliana Mello, from the Faculty of Arts of the Federal University of Minas Gerais [23, 24, 16]. This synchronic corpus was recorded between 2008 and 2011 and is composed

[16] `https://librivox.org/`

[17] `http://www.openslr.org/94/`

[18] www.openslr.org/100

[19] `https://www.alip.ibilce.unesp.br/`

[20] `https://www.alip.ibilce.unesp.br/termos-de-uso`

[21] `http://www.c-oral-brasil.org/`

of informal and spontaneous speech, representative of the linguistic variation in Minas Gerais, especially in the city of Belo Horizonte.

It is composed of 139 recording sessions (or texts), totaling 21.13 hours and 208, 130 words, averaging 1,500 words per text. C-ORAL Brasil I has 362 informants. There is a balance regarding number of uttered words: 50.36% words are uttered by 159 males and 49.64% words by 203 females.

Its content is divided into private-family (about 3/4 of the corpus) and public (1/4) contexts. In addition, there is a separation of interaction types by number of participants: monologues (amounting to about 1/3 of recordings), dialogues and conversations, i.e. more than two active participants (about 2/3 of recordings).

The speech flow was segmented into tonal units and terminal units according to the prosodic criterion, based on the Language Into Act Theory (L-AcT) [25] which designates the utterance as the reference unit of speech. The boundary between tonal units results from a prosodic break with a non-conclusive value, while the boundary between terminal units corresponds to the perception of a prosodic break with a conclusive value.

In order to obtain a great diaphasic diversity, i.e, according to the communicative context, the project brought a remarkable variety of communicative contexts, compiling scenarios such as communication between players in a football match, the preparation of a drag queen for a presentation, a conversation between a realtor and a client, among others. In addition, a considerable balance was reached regarding the demographic criterion concerning the informants' education and gender. There are 362 informants in the corpus, 138 from the city of Belo Horizonte, 89 from other cities in Minas Gerais, and the rest from other states, countries, or of unknown origin.

There was an effort to use high-quality acoustic equipment at the time. The project used PMD660 Marantz digital recorders and Sennheiser Evolution EW100 G2 wireless kits. It also used non-invasive "clip-on" microphones to create a more natural environment, essential for recording high diaphasic variation in spontaneous speech.

C-ORAL Brasil I is available via download from the project website in raw format, morphosyntactically annotated by the Parser Palavras [26], in addition to metadata. The C-Oral-Brasil I corpus is licensed under CC BY-NC-SA 4.0. The following files are of special interest for this work: (i) audio in .wav format, with a sampling rate of 48kHz, transcription in .rtf and .txt formats, audio-transcription alignment in XML format generated by the software WinPitch[22].

2.2.3. NURC-Recife

The NURC-Recife corpus has its origins in the 1969 NURC (*Norma Urbana Oral Culta*) project, which documents the spoken language in five Brazilian capitals: Recife, Salvador, Rio de Janeiro, São Paulo and Porto Alegre. NURC-Recife corresponds to the part referring to the linguistic variety spoken in the city of Recife. The corpus is available on the website of the NURC Digital project[23], developed between 2012-2016. The project NURC Digital, coordinated by Prof. Miguel Oliveira Jr. of the Federal University of Alagoas (UFAL), was responsible for processing, organizing and releasing the data of the NURC-Recife project in digital form [17].

The project is comprised of 290 hours spread over 346 recordings (called inquiry in the project) obtained between the years of 1974 and 1988. In fact, this value would be the total duration in hours if all audios and their transcriptions were available on the website. An analysis of all audio-transcription pairs raised one inquiry lacking audio and transcription and 11 inquiries lacking transcriptions, resulting 279 hours available.

The recordings follow NURC guidelines and are categorized as follows:

- Formal utterances (EF), consisting of 37 recordings of lectures and talks given by one speaker;
- Dialogues between two informants (D2) conducted by a mediator, with 71 recordings;
- Dialogues between an informant and an interviewer (DID), with 238 recordings.

The informant ages range from 25 to over 56 years, all of them with higher education and initially selected with equal division (originally 300-300) for male and female voices.

The environment of the recordings varied, depending on the type of inquiry: specific rooms, classrooms, auditoriums or even in the informants' homes. It also has very heterogeneous noise levels and sound clarity, whether from the equipment used, the recording environment or deterioration of the physical material.

The original recordings were captured with omnidirectional dynamic microphones with table support. The reel-to-reel tape recorders used were: AKAI 4000 DS Mk–II, SONY TC–366, and Philips N 4416, the first being the most frequent. The audios were recorded on professional reel magnetic tapes, 0.0018mm thick, 6.35mm wide, and 540m long (BASF TP 18 LH). However, within the scope of the NURC Digital project, they were digitized following the recommendations of the Open Archival Information System (OAIS), in the ISO standard (14721 : 2003), with a sampling rate of 96kHz and quantization of 24 bits. For this digitization, were used the software Audacity, Audiofile Specter, the AKAI 4000 DS Mk–II reel-to-reel recorder, a USB Audio Interface Sound Devices USBPre 2, and the RCA Diamond Cable JX-2055.

NURC Digital is available for academic use, without a defined license, via download from the project website, which allows a search by recording year (1974 to 1988), recording topic, and type of inquiry (D2, DID, and EF). There is also information about the age range of the informants, gender, and audio quality. Within each inquiry folder there are: (i) the digitized version of the specific recording (metadata), in .pdf format; (ii) a file in textgrid format, containing the audio timestamps with the transcriptions; (iii) the audio file of the recording in .wav format (48kHz); (iv) a copy of the audio file, also in .wav format, compressed at a frequency of 44kHz; and (v) the original transcription in .pdf format.

2.2.4. SP2010

The SP2010 project [27, 18] was coordinated by Prof. Ronald Beline Mendes, of the Research Group in Sociolinguistics at FFLCH/USP (GESOL-USP). It started in 2009 and ended in 2013 to document and study the Portuguese spoken in the city of São Paulo. The project was supported by the FAPESP agency between 2011 and 2013, generating a corpus publicly available for academic research.

The corpus contains 60 recordings of 60 to 70 minutes each, collected between 2012 and 2013[24], with equal division for fe-

[22] https://www.winpitch.com/

[23] https://fale.ufal.br/projeto/nurcdigital/

[24] http://projetosp2010.fflch.usp.br/corpus

male and male voices. Each recording identifies an interview with an informant, comprising two parts:

- an informal and spontaneous conversation, with questions about the informant's neighborhood, family, childhood, work and leisure, seeking personal involvement;
- the continuation of the conversation, but exploring a more argumentative speech, with questions on more objective themes about the city of São Paulo, involving problems, solutions, characterizations of the city and its inhabitants. In addition, there are three reading recordings: a list of words, a news article and a statement. Finally, specific questions about the sociolinguistic varieties of the city are proposed.

The informants were selected to represent 12 sociolinguistic profiles characterized by distinct combinations of the following variations: age group, (with three age groups encompassing individuals from 19 to 89 years), education, (with two school stages represented — up to elementary school and with higher education), and gender, (male and female). Each sociolinguistic profile has five informants as representatives, each with a recording. The informants' region of residence within the city was also considered, and a balance of informants was sought in this regard, considering the division of São Paulo into 3 regions: *Centro Velho, Centro Expandido* and *Periferia*.

For the recording, the authors used TASCAM DR100 MK2 digital recorders and Sennheiser HMD25-1 microphones, having varied recording conditions, with some interviews being more noisy than others, as they were not conducted in specialized and isolated environments.

The material collected in the SP2010 project is made available via download from the project website, free of charge to the academic community of researchers. Eight files are available for each interview: two audio files — in .wav stereo format, 44kHz, and also in .mp3; four transcription files (in .eaf, .doc, .txt and textGrid formats); the informant and the recording forms (in .xls format); and a .zip file that contains all of the interview materials except the .wav file.

2.2.5. TEDx Portuguese

TEDx Portuguese is a new corpus compiled specifically for CORAA v1. It should not be confounded with the BP audios available in Multilingual TEDx Corpus (described in Section 2.1). TEDx Portuguese is based on the TEDx Talks[25], which are events in which presentations on a wide range of topics take place, and in the same format as the TED Talks[26], but in languages other than English.

Although they are independent meetings, they are licensed and guided by the TED organization, that is, they are short presentations, containing prepared speech, with a duration recommendation of less than 18 minutes, typically presented by a single presenter. The "x" at the end indicates that the event is carried out by autonomous entities worldwide. More than 3,000 new recordings are made annually[27].

To create this dataset, we selected presentations spoken in Portuguese, both from Brazil and Portugal, with available preexisting subtitles. After selecting the presentations, they were downloaded, were the audios were extracted and converted to .wav format, mono, with a sampling rate equal to 44kHz. BP presentations have accents from practically all regions of Brazil.

The subtitles were also downloaded, with the text extracted exclusively, that is, the timestamps were discarded. The dataset is composed of excerpts from 908 talks (671 of which are in BP), totaling at least 908 different speakers, since there are also talks with more than one speaker. The variant (PT-PT or PT-BR) is annotated in the dataset metadata. Considering both variants, there are 543 male and 375 female voices.

3. Data Processing Pipeline

In this section, we present the processing steps of the CORAA corpus:

1. Normalization of transcriptions,
2. Segmentation and removal of silence and untranscribed parts of speech,
3. Forced alignment between audio and corpora transcription for two corpora[28],
4. Specific processing in the ALIP and NURC-Recife corpora. For example, (i) to maintain the capitalization of letters indicative of names, to aid in the expansion of names, (ii) to preserve the slashing annotation, indicative of truncation in the speaker's speech, to aid in the identification of truncated audios, and (iii) to discard audios with duration less than 0.3 seconds in the NURC-Recife[29],
5. Validation of audio-transcription pairs, via the web interface created in the project, so that the CORAA v1 corpus can be used for training ASR methods, and
6. Evaluation of agreement between annotators and between annotators and the gold-standard annotation, performed by a trained annotator.

All corpora described in Section 2.2 were obtained from their respective official websites. After downloading, all transcripts were converted to .csv format and the organization of audio files was standardized. Additionally, due to the differences between the transcription rules of each corpora, text normalization was performed, described in Section 3.1. Furthermore, as the ALIP corpus does not originally have alignment between the transcription and the audio file, we performed the forced alignment between the transcription and the audio. TEDx Portuguese has the alignment provided by the subtitles, however, this alignment is limited to 42 characters per line to optimize screen display, and may not correspond to sentence boundaries, for this reason we also performed forced alignment in TEDx Portuguese. We describe the forced alignment process in these two corpora in Section 3.2. The validation of the audio-transcription pairs is presented in Section 3.3 and the evaluation of agreement between annotators and between annotators and the gold standard annotation is presented in Section 3.4. Finally, Section 3.5 presents the statistics of the five corpora that make up CORAA, after its pre-processing.

3.1. Text Normalization

The four academic project corpora used their own transcription criteria. The oldest and most widely cited transcription

[25] `https://www.ted.com/watch/TeDx-talks`

[26] `https://www.ted.com/`

[27] `https://www.ted.com/about/programs-initiatives/TeDx-program`

[28] ALIP was not available in an aligned way and TEDx Portuguese were available with segmentation to optimize on-screen presentation

[29] The original duration of the corpus (279 hours) dropped to 216 hours.

standards are those of the NURC Project, which were used by NURC-Recife. NURC-Recife follows the orthographic transcription and its rules can be found in [28]. During the NURC Digital project, NURC-Recife went through new processing steps, including: quality verification of digitized audio, manual alignment between audio and transcription, spelling revision using a spell checker, which are described by [17].

The corpus C-Oral-Brasil I follows the orthographic-based transcription criteria, but with the implementation of some non-orthographic criteria to capture grammaticalization or lexicalization phenomena [29]. For example, there are aphereses (disappearance of a phoneme at the beginning of a word), reduced prepositions, absence of plural mark in noun phrases, cliticizations of pronouns and pre-verbal negation and articulations of preposition with article.

The SP2010 project uses semi-orthographic transcriptions, using the following criteria: (i) no change in the spelling of words, as phonetic transcription is not used; (ii) no grammatical corrections; (iii) use of parentheses to indicate the deletion of /r/ in syllabic coda, syllable /es/ of the verb "estar" (to be), in all tenses and verb modes, and syllable "vo" of "você(s)" (you). Other deletions were not indicated with marks. Filled pauses, interjections, and conversational markers such as "right ?", "okay ?" were pervasively used.

The ALIP project follows the orthographic conventions of the written language, but uses capital initials only for proper names. The transcription annotates the following variable phenomena [15]: (i) vowel raising in contexts of medial postonic of nouns, as in "c[o]zinha $\sim$ c[u]zinha" and of verbs, as in "d[e]via $\sim$ d[i]via"; (ii) postonic lifting and syncope medial, as in "pes.s[e].go $\sim$ pes.s[i].go $\sim$ pes.go"; (iii) gerund reduction, as in "canta[ndo] $\sim$ canta[no]", a striking feature of São Paulo speech.

Results for variable phenomena of morphosyntactic order include, for example, the realization of prepositions with and without contraction, as in "com a $\sim$ cu'a $\sim$ c'a", "para $\sim$ pra $\sim$ pa". The corpus proposed a transcription system based on the NURC project and reports the transcription conventions grouped in the following criteria: (i) word spelling, which includes, for example, question and exclamation marks next to the markers discursive and interjections, use of "/" for word truncations; (ii) prosodic elements where it uses an ellipsis for pauses, double-typed colons for lengthening vowels, and interrogation for questions; (iii) interaction in which it identifies the participants of the interaction and use square brackets for voice overlappings; (iv) transcriber's comments where parentheses are used for hypotheses of what is heard and double parentheses for descriptive comments for laughs, for example.

Considering these differences between the transcriptions and seeking to maintain standardization, we performed the following normalizations in the texts of all CORAA corpora. Some normalizations were performed before validation (items (1), (2), (3)) and practically the entire list below was performed at the end of the entire process, since the ALIP and TEDx Portuguese corpora had their transcriptions revised:

1. Removal of extra annotations that do not belong to the alignment of transcripts and audios, such as annotations that indicate the speech of the interviewer and interviewee, truncations, laughter and extra information provided by the annotators of the projects that make up CORAA corpus;
2. Normalization of texts to lower case;
3. Removal of duplicate spaces;
4. Expansion of acronyms for their forms of pronunciation (standardization applied after validation, to guarantee the expansion of all acronyms);
5. Standardization of some uses of filled pauses, using a reduced set of these: *ah*, *eh* and *uh*. Some variations of these representations have been replaced by the closest of the three above (e.g.: *hum, hm, uhm* was replaced by *uh*; *éh, ehm, ehn*, was replaced by *eh*; *huh, uh, ã* was replaced by *ah*);
6. Expansion of cardinal and ordinal numbers, using the num2words library[30];
7. Percentage sign expansion (%) for its transcribed form (percentage);
8. Removal of characters such as punctuation and non-language symbols (such as parenthesis and hyphen).

It is important to note that the corpus also brings a great variety of filled pauses forms, so that the model can learn to vary its use, although this richness penalizes the evaluation of models trained with CORAA v1 corpus, as detailed in Section 4.3.

3.2. Automatic Forced Alignment

As mentioned before, in the ALIP and TEDx Portuguese corpora the alignment between the transcripts and audio was performed using an automatic forced alignment method. For this, we use the tool Aeneas[31]. This tool requires the text segmented into sentences or excerpts.

In the ALIP corpus, the text was segmented using the annotations of pauses or hesitations, indicated by ellipses ("...") and turn-shifts between speakers, indicated by a line break followed by the next speaker identification abbreviation, present in the original annotated corpus.

In the TEDx Portuguese corpus, the segmentation of text into sentences was performed using the punctuations present in the subtitles, if any. For this, a maximum limit of 30 words was defined for each sentence and, when this limit was reached, the sentence was divided in the point before this limit. In the case of no punctuation, the sentences were divided in an arbitrary way, for example, in silent passages, or with music, or based on variations in speech rate.

3.3. Human Validation via Web-based Platform

The validation of audio-transcription pairs was performed in a simple web interface through two tasks: **binary annotation (VALID - INVALID)** and **transcription** to correct automatic alignment effects, as was the case with ALIP corpus, or to review manual transcripts, previously made, as was the case for the TEDx Portuguese corpus.

The **binary annotation** was carried out by: listening to an audio file that could be listened to as many times as necessary and the reading of the original transcription. The annotation was binary, that is, the pairs were classified as valid or invalid, and it was necessary to point out the reason for such choice, which provided a guide for the choice itself.

There are 3 main reasons an audio is considered invalid:

1. Voice overlapping;

[30] `https://github.com/savoirfairelinux/num2words`

[31] Available at `http://www.readbeyond.it/aeneas`.

2. Low volume of the main speaker's voice, making the audio incomprehensible;
3. Word truncation.

There are also 3 causes for considering a transcript as invalid, i.e. when it is not aligned with the audio, because there are:

1. Too many words in the transcript;
2. Too few words;
3. Words swapped.

The following options were given to validate an audio/transcript pair:

1. Valid without problems.
2. Valid with filled pause(s).
3. Valid with hesitation.
4. Valid with background noise/low voice but understandable.
5. Valid with little voice overlapping.

In cases where there is an audio with hesitation but the transcription does not correspond to the pauses made, the pair must be invalidated. After one pair has been annotated, another is provided and this process continues until the user wants to stop the annotation and/or disconnect.

In the web interface for validation, the **transcription task** has a screen composed of the original transcription, a player for the audio file that can be repeated as many times as necessary, an editing window initially filled with the original transcription, which is used by the annotator to transcribe, and a button to send the transcription. To complete the task of transcribing an audio, the annotator must listen to the audio.

The annotator must also analyze if this audio fits into any of the types below: music, clapping, word truncation in the audio, loud noise or another language other than Portuguese, very low voice, incomprehensible voice, foul words, hate speech, and loud second voice. If so, the annotator should insert the symbols "###" (denoting invalid audio) in the edit window and send its response. As we focused on the BP, we decided to kept 4.69 hours of European Portuguese, so during most of the project, annotators were instructed to discard European Portuguese audios.

The annotators were instructed to comply with the following eight guidelines:

1. Do not change to the grammar normative form the following signs of orality in the audio: "tá/tó, né, cê, cês, pro, pra, dum, duma, num, numa".
2. Transcribe filled pauses, such as "hum, aham, uh" as heard.
3. Transcribe repetitive hesitations such as "da da", or "do do" as heard.
4. Write numbers in full form.
5. Letters that appear alone should be spelled out.
6. Acronyms and abbreviations should be transcribed in full form, using the English alphabet for those in English and the Portuguese alphabet for those that appear in Portuguese.
7. Foreign words should be transcribed normally, in the language in which they appeared.
8. Punctuation and case sensitivity could be applied, as normalization is performed in post-processing phase.

3.4. Kappa Evaluation: subjectivity of the Human Annotation

The validation of audio-transcription pairs of the CORAA v1 corpus, using the binary annotation and transcription tasks (see Section 3.3), was performed from October 2020 to July 2021, when it was generated the database export.

The number of annotators varied during the project duration. In total, 63 different annotators performed the validation, which could be divided into 4 main annotation groups according to the start and end dates of each annotator on the project. Two groups validated the corpora for 3 months in 2020 (October to December), with some annotators in this group continuing the validation in 2021. There was a 1-month annotation task-force during December 2020. The final group started the CORAA v1 validation work in May 2021 and ended in July 2021.

Each group attended a lecture on the validation process, read the tutorials for the two tasks (annotation and transcription) and received instructions to ask elucidate doubts via the project email throughout the process.

At the beginning of the validation process, from October to December 2020, each audio-transcription pair was annotated by two or three annotators, so that we could use the majority vote to export the data, discarding the divergent pairs, in this initial phase of learning how to validate. Agreement between annotators was calculated in two ways: between annotators who annotated the same pairs (Section 3.4.1) and based on a gold-standard annotation of samples from all datasets, performed by a project member (Section 3.4.2).

3.4.1. Kappa among Annotators

Two Fleiss kappa values were calculated for the annotation from October to December 2020, to separate the groups of annotators. The project started with two groups in October, totaling 28 annotators, but with the entry of a new group on November 23, 2020 the number of annotators went to 63. Thus, it was decided to calculate a kappa value to evaluate each period of annotation — from October 1st to November 23rd and from November 24th to December 31st, 2020. The hypothesis was that the annotation would become easier and with high agreement as the practice increased. However, there is another variable that influenced the agreement: the different transcription rules for each corpus of the CORAA corpus (see Section 3.1) also influenced the agreement. We calculated the agreement value via Fleiss' kappa twice, once considering only two annotators and the other considering only three annotators, according to the total number of annotators of a given audio. The values are shown in Table 2.

It is observed that there are absent values on the table, because the specific corpus was not being annotated in the referred period. The great disagreement between the annotators showed a more subjective task than previously imagined. By manually comparing audios in which annotators agreed with audios in which they disagreed, some points became clear: (i) the human ear naturally tends to complete truncated words, so that different annotators may disagree in defining whether an audio is in fact truncated or not, (ii) background noise level and voice pitch (low/high) are very subjective concepts, and different people are expected to consider different noise levels as tolerable, (iii) naturally, due to the ease of understanding different accents, annotators from different regions of the country tend to understand more or less of the audio according to the their accent, which can also be a source of disagreement.

Table 2: *Kappa values for each dataset in two annotation periods, separated by number of annotators. In the last three months of 2020, the order of validation of the corpora was C-Oral-Brasil I, SP-2010 and NURC-Recife.*

	1/10 - 23/11		24/11 - 31/12	
	2 annotators	3 annotators	2 annotators	3 annotators
Number of pairs	6,785	29,835	26,974	4,224
Number of annotators	25	25	51	51
Kappa Values				
C-ORAL Brasil I	0,394	0,353	—	—
SP-2010	0,420	0,394	—	—
NURC-Recife	—	—	0,317	0,314
Total	0,391	0,392	0,317	0,314

3.4.2. *Kappa for the gold-standard annotation*

The gold standard was built to maintain the representativeness of all validated corpora, and all participating annotators, according to the following process:

1. For each annotated corpus, we generated a list of all annotators in that corpus;
2. For each name present in the list, five pairs annotated by the annotator were randomly selected (annotators with less than 5 pairs annotated per corpus had their pairs discarded);
3. The selected pairs were duplicated and annotated by an experienced annotator of the project, creating a gold-standard annotation with the following distribution:
   - **Alip**: 15 annotators and 75 pairs
   - **C-ORAL Brasil I**: 24 annotators and 120 pairs,
   - **NURC-Recife**: 55 annotators and 275 pairs,
   - **SP-2010**: 25 annotators and 125 pairs,
   - **TEDx Portuguese**: 50 annotators and 250 pairs.
   - **Total**: 845 pairs (520 from the binary annotation task and 325 from the transcription task)

The consensus pairs between the annotators were included in the exported dataset, that is, if the absolute majority chose to validate the pairs. Thus, we analyzed the degree of agreement of the annotators together (exported values) in comparison with the gold-standard corpus. The value obtained was **0.514**, showing a "moderate agreement", according to [30]. Even though the task is subjective, the final result obtained from the annotation of the exported pairs was satisfactory.

3.5. Datasets Statistics

Overall, CORAA has 290.77 hours of validated audios, containing at least 65% of its contents in the form of spontaneous speech. We will refer as the processed version of the corpora in CORAA as sub-datasets. NURC-Recife sub-dataset includes conference and class talks, considered prepared speech (see Table 1). Currently, no other dataset for BP includes audios in this speaking style. Therefore, the task of ASR is more challenging than for other datasets. Another CORAA characteristic is the presence of noise in some of its sub-datasets, which is also more challenging for models created for this task. Table 3 presents statistics for each validated sub-dataset in CORAA v1. The resulting set encompasses almost 1,700 speakers.

Audio durations range, in average, for 2.4 to 7.6 seconds according to sub-dataset. Audios having more than 200 words or 40 seconds were automatically filtered from the dataset. Figure 1 presents estimated speaker distribution in each sub-dataset according to sex. Overall, the distribution is similar for males and females[32]. Figure 2 presents audio duration distributions by sub-dataset. The audios are ranked by duration and their relative position (percentil) is shown in the x axis. Audios duration are presented in the y axis. Percentils are used to simplify sub-dataset comparisons. Figure 3 is similar, but presenting word distribution per dataset.

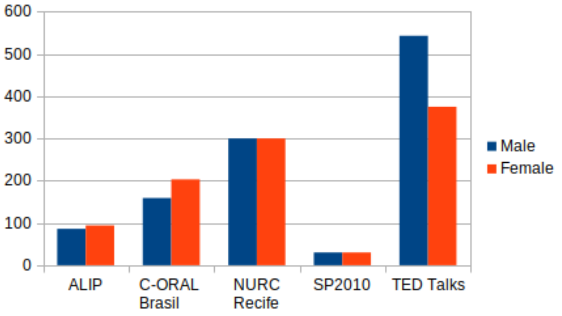


Figure 1: *Estimated Speaker Distribution by Sex*

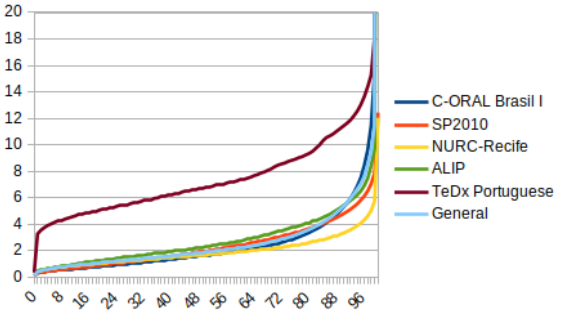


Figure 2: *Duration distribution per sub-dataset*

Regarding duration, the segmentation process play a role in the obtained durations. Only ALIP and TEDx Portuguese were automatically segmented. The other sub-datasets were manually segmented. For the automatic segmentation, the param-

[32] In the corpus C-ORAL Brasil I, there is a balance regarding number of uttered words — 50.36% words are uttered by 203 females and 49.64% words are uttered by 159 males

Table 3: *Statistics for each processed version of the projects included in CORAA v1 (hours in decimal)*

Corpus	**ALIP**	**C-Oral Brasil I**	**NURC Recife**	**SP2010**	**TEDx Port.**	**Total**
Original (hrs)	78	21.13	279	65	249	692.21
Validated (hrs)	35.96	9.64	141.31	31.14	72.74	290.79
BP Speakers	179	362	417	60	671	1,689
Audios (segmented)	45,006	13,668	261,906	46,482	35,404	402,466
Audio Duration (sec.)	2.90	2.46	1.94	2.43	7.55	3.39
Avg Tokens	53.910	60.079	20.418	48.118	166.369	41.546
Avg Types	6.391	7.188	3.733	6.002	8.807	5.581
Total Tokens	335,664	99,954	1,378,558	339,890	610,639	2,764,705
Total Types	14,189	8,715	41,903	12,351	27,469	58,237
Type/Token Ratio	0.042	0.087	0.030	0.036	0.046	0.022

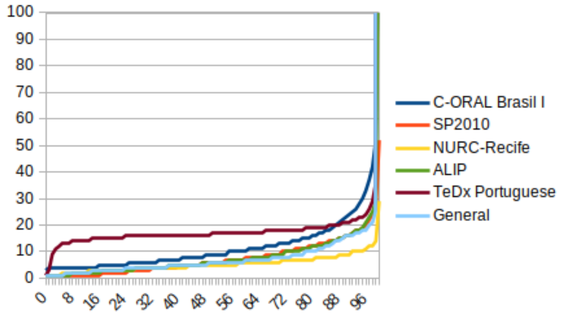


Figure 3: *Word distribution per dataset*

eters were adjusted aiming at better segmentation of informational units. ALIP had a similar duration than the others dataset. However, TEDx Portuguese audios tended to be longer. Speech style and genre also play a role in the obtained results. When pronounce is faster and with less pauses, there are less places in the audio that the segmentation software is confident to break the utterances. TEDx Portuguese is the main source of prepared speech in CORAA and had the longest audios and the same applies to word distribution, which is natural since the audios are longer. The remaining corpus presented similar distributions among them.

4. Baseline Model Development

We performed a experiment over CORAA Dataset in order to measure the dataset quality, potentials and limitations. Before the execution of this experiment, the dataset was divided into three subsets: train, development and test. Table 4 presents the approximate number of hours for these sets for each subdataset, as well as the number of speakers from each sex. Subdataset validation sets were adjusted to have approximately 1 hour. Test sets were built in a similar manner, but having approximately 2 hours. This decision is supported by the work of [31], which recommends that test sets should have at least 2 hours. NURC-Recife test set contains more than 3 hours of audios, because this sub-dataset have more speech genres than the others. All the audios from European Portuguese were included in the training set.

4.1. Experiments

Our proposed experiment is based on the work of [21]. These authors fine-tuned the model Wav2Vec 2.0 XLSR-53 [19, 20] for ASR, using public available resources for BP. One of their experiments consisted on training 437.2 hours of Brazilian Portuguese. Wav2Vec 2.0 is model that learns quantized latent space representation from audios by solving a contrastive task. First, the model is pre-trained using an unsupervised approach in large datasets. Then, it is fine-tuned for the ASR task using supervised learning. Wav2Vec XLSR-53 is pre-trained over 53 languages, including Portuguese.

In our approach, Wav2Vec XLSR-53 is fine-tuned for CORAA v1. We also evaluated [21] public fine-tuned model against CORAA v1, using the sets presented in Table 4.

Using the proposed training, development and testing divisions for CORAA v1, we explores training Wav2Vec 2.0 XLSR-53 model using CORAA v1 during 40 epochs. Similarly to the work of [19] and [21], we opted to freeze the model feature extractor.

To train the model, we use the framework HuggingFace Transformers [32]. The model was trained with GPU NVIDIA TESLA V100 32GB using a batch size of 8 and gradient accumulation over 24 steps. We used the optimizer AdamW [33] with a linear learning rate warm-up from 0 to 3e-05 in the first two epochs and after using linear decay to zero. During training the best checkpoint was chosen, using the loss in the development set. The code used to perform the experiment as well as the checkpoint of the trained model are publicly available at: `https://github.com/Edresson/Wav2Vec-Wrapper`.

4.2. Results and Discussions

Section 4.2.1 presents a comparison of our results with the work of [21]. The models are tested against the entire test subset of CORAA v1 and Common Voice version 7.0 (Portuguese audios). Therefore, our model is evaluated *in-domain* using CORAA v1 test set, a dataset in which it was fine-tuned for specific recording characteristics. At the same time, our model is also evaluated *out-of-domain* in Common Voice, a dataset completely new to our model.

Additionally, Section 4.2.2 focuses on evaluating the models in test sets of CORAA sub-datasets. This enables a more detailed analysis on factors such as audio quality and accents.

Table 4: *Statistics of Train/Dev/Test partitions of each CORAA corpora.*

	Duration (hrs)			**Num. Speakers (M—F)**		
Subset	Train	Dev	Test	Train	Dev	Test
ALIP	33.40	0.99	1.57	80—87	2—2	4—5
C-ORAL Brasil I	6.54	1.13	1.97	138—181	9—9	12—13
NURC-Recife	137.08	1.29	2.94	295—296	2—1	3—3
SP2010	27.83	1.13	2.18	27—27	1—1	2—2
TEDx Portuguese	68.67	1.37	2.70	532—364	4—4	7—7
Total	273.51	5.91	11.35	1072—955	18—17	28—30

Finally, Section 4.2.3 investigates the two speech styles: prepared or spontaneous.

4.2.1. In/Out of Domain Evaluation

Table 5 presents the comparison of our experiment with the work of [21]. First, we performed an in-domain analysis of our model using CORAA v1 test set. Then, our model is evaluated out-of-domain using Common Voice test set. It is important to observe that, for the compared work, the analysis is mirrored, there is, CORAA v1 is the out-of-domain evaluation and Common Voice is the in-domain analysis.

In the Common voice dataset, as expected, [21] model performed better. Regarding WER, it can be noted that our model is less than 7% above their work. We also focuses our analysis on the metric CER, because for smaller audios, with just a few words, this metric tends to be more reliable. In this scenario, our models are approximately 2% worse than the model from [21]. On the other hand, in the CORAA dataset, our model presented a much superior performance (more than 19% in WER and 11% in CER). Furthermore, our experiment managed to generalize better for audio characteristics not seen during training, achieving an average higher than the performance of the [21]. This is very interesting especially because the [21] model was trained with approximately 147 hours of speech more than our model.

We believe that models trained with the CORAA v1 dataset generalize better than a model trained with existing publicly available datasets for BP due to the spontaneous speech phenomenon and the wide range of noise and different acoustic characteristics present in CORAA. Furthermore, accent can be a factor since the datasets used in the training of the [21] model may not cover in depth all accents present in the CORAA v1.

4.2.2. Sub-dataset Analysis

There are important differences in the recording environment for each sub-dataset. Additionally, they also varies on accents. Table 6 presents the test performance for each CORAA v1 sub-dataset.

Regarding datasets, ALIP presented the greatest challenge for the models, both for CER and WER metrics. We believe this occurred because audios from ALIP presented more noise than the other sub-datasets.

Regarding accents, we have different results. On one hand, our model presented similar performances in NURC-Recife and SP2010, which have two distinct accents (Recife and São Paulo city). On the other hand, C-ORAL Brasil presented higher WERs and CERs than the other two. Two factors may have influenced this result. First, audio quality and noise presence tend to play a major role in model performances. Second, C-ORAL Brasil accent (Minas Gerais) has two characteristics that are difficult for models: speech rate is faster and there is more word agglutinations. As a consequence, the analysis was inconclusive for this accent, since the results are influenced both by the accent and the speech rate.

Regarding experiments, our model presented results varying from 19 to 34% in WER and from 7 to 17% in CER. On the other hand, [21] presented higher error rates, which is expected considering the training of their model had no previous contact with CORAA v1 audios.

4.2.3. Spontaneous vs Prepared Speech Analysis

Table 7 presents an analysis in which sub-datasets are merged according to speech style. The Spontaneous Speech column is obtained from the merging of ALIP, C-ORAL Brasil I, SP2010 and parts of NURC-Recife. The prepared speech column contains TEDx Portuguese and parts of NURC-Recife. As expected, the models perform better on prepared speech. However, for several ASR applications, spontaneous speech is more relevant (for example, ASR of phone call and meetings). This can also observed in Section 4.2.2, as TEDx Portuguese presented the lowest error rates.

4.3. Error Analysis

The current test dataset is composed of 13,932 audio-transcription pairs, totaling 11.63 hours (see Section 4), with parts from all CORAA v1 dataset.

As this is the first time that a dataset composed of spontaneous speech samples was used to train an ASR model for BP, we performed a more detailed analysis of the errors from our model in a sample of the test dataset.

The 13,932 test pairs were ordered by the CER values of our model to illustrate the different types of errors and to analyze whether there is a relationship of error types with CER values. The automatic transcription was analyzed using the typology of [34], adapted for the task of evaluating ASR models.

The typology used here to illustrate the model errors is composed of 11 error types, grouped into 6 more general classes: Alphabetical, Lexical, Morphological, Language and Spontaneous Speech, Semantic, and Diacritic Placement Errors. Below we present a description of the 11 errors with examples.

- Alphabetical errors are alphabetic writing application errors.
  1) Alphabetical errors occur in 3 situations: by transcribing speech directly into writing, in complex syllables or even with ambiguous letters (“ce” versus “sse” or “sa” versus “za”, in Portuguese).
  An example of this type of error is related to the sound /k/ in Portuguese which is represented by the letter “c” before some vowels and by “qu” before other vowels. Thus, the use of “c” in place of “qu” is associated with the speaking/writing relationship.

Table 5: *Results for the In/Out of Domain Analysis.*

Datasets	**Common voice**		**CORAA**		**Mean**	
Metric	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**
[21]	**4.15**	**13.85**	22.32	43.7	13.23	28.77
Our	6.34	20.08	**11.02**	**24.18**	**8.68**	**22.13**

Table 6: *Results in the CORAA test set for all subsets.*

Datasets	**[21]**		**Our**	
	CER	**WER**	**CER**	**WER**
ALIP	33.72	59.30	17.30	34.06
C-ORAL Brasil I	23.53	45.9	13.62	28.88
NURC-Recife	19.46	42.17	9.09	22.03
TEDx Portuguese	9.75	22.69	7.43	19.36
SP2010	23.11	42.44	9.57	20.00

Table 7: *Results Spontaneous vs Prepared Speech.*

Speech Style	**Spontaneous Speech**		**Prepared Speech**	
Metric	**CER**	**WER**	**CER**	**WER**
[21]	25.75	49.18	**5.30**	**15.89**
Our	**12.44**	**26.5**	6.07	18.7

- Lexical errors occur in an excerpt transcribed by the ASR where there is:
  2) omission or addition of words;
  3) exchange of words.
  An example from our dataset regarding addition of a word in the automatic transcription is “que legal” instead of “legal”
  Also from our dataset, an example of word exchange is “e que mais que a gente vida” instead of “e que mais que a gente viu”.
- Morphological errors are errors that occur due to the violation of writing rules that is linked to the morphological structure of words. These are errors from:
  4) omitting morphemes (e.g. “come” written instead of “comer”);
  5) concatenation of morphemes (e.g., “agente” instead of “a gente”, or “acasa” instead of “a casa”);
  6) separation of morphemes, as in the example: “de ele” written instead of the contraction “dele”).
- Language and spontaneous speech errors are errors of:
  7) Words in English (or in a language other than BP) wrongly transcribed;
  8) Filled pause errors (e.g., “é” versus “eh”) where the transcription and model responses diverge;
  9) Spontaneous speech errors (e.g., “tá” versus “está”; “té” versus “até”; “cê” versus “você”) in which transcription and model responses diverge.
- Semantic errors occur when two words are spelled similarly but have different meanings.
  10) Semantic errors (e.g. “Ela comprimentou o diretor assim que chegou.”, where the correct form would be “cumprimentou”).
- Diacritic placement errors occur due to missing accents or improperly adding them. They are problematic because the five training corpora were built at different times, in which there were different spelling rules for the Portuguese language. For example, the last orthographic agreement for the Portuguese language came into force in Brazil in 2016.
  11) Accent marks errors.

Table 8 shows examples of 11 errors presented above (column 1), in which the original transcript (column 2) and the model response (column 3) diverge. The location of the error in the snippets appears in bold.

A sample of 938 audio-transcription pairs was analyzed, of which 134 contained some errors in the audio transcription and thus they were not framed in the typology. Also, 309 pairs were annotated for deletion as their audio were compromised (because of truncation, very loud noise or overlapping voices). The remaining 500 pairs, according to the CER ranges analyzed, are shown in column 1 of Table 9. They were categorized according to the typology presented above. For some pairs more than one error occurs and for some excerpts with high CER values only one error was annotated (the most frequent) although the transcription had many more.

This initial analysis has already resulted in a decision to make a revision in all pairs of the test dataset, which is currently being conducted, and should result in a new version CORAA in the future.

Table 9 shows, in the last column, the variety of error types in each range presented column 1; its frequency is shown in parentheses. We present in bold the most frequent type.

The lexical error of type 3 — exchange of words — is the most frequent one, which is expected given that the task is automatic transcription, and the training process of these models favors the recognition of frequent and well formed words. Moreover, omission and addition of words (error type 2) is pervasive as it appears in all the intervals (even in the last one, where CER varies from 0.7 to 12, although it was not explicitly annotated). However, the second and third errors classified by frequency are: concatenation and filled pause swap error. The latter is related to the fact that the CORAA dataset has a large percentage of spontaneous speech samples in which both the number and variety of filled pauses are high.

Table 8: *Examples of the 11 different error types.*

Error Type	Original Transcription	ASR Transcription
1	uma maneira de saber o que e como o indivíduo **identifica** algo	uma maneira de saber o que e como o indivíduo **identifiqa** algo
2	ou pra dar um apoio moral	ou pra dar um apoio **im** moral
3	o outro **foi** morar um pouco mais longe	o outro **prai** morar um pouco mais longe
5,4	**que lhe dão** ora **dor**	**ciridão** ora **do**
5	criança é mais **coca cola*** biscoito	criança é mais **cocacola** biscoito
6	que levaria a uma resposta **aquele** estímulo ah legal faz **tempão** já	que levaria a uma resposta **a quele** estímulo ah legal faz **tem pão** já
7	na teoria de **osgood** é que de **jazz**	na teoria de **osguot** é que de **dez**
8	e essa daí **eh** **eh** **eh** **ham**	e essa daí **é** **é** **ahn** **uhn**
9	**pra** área específica que é o curso diz que é um curso excelente	**para** área específica que é o curso diz que é um curso excelente
10	entendeu era eles **suavam** mais a camisa pelo clube entendeu e	entendeu era eles **soavam** mais a camisa pelo clube entendeu e
11	então **é** conhecer a população usuária do equipamento urbano	então **e** conhecer a população usuária do equipamento urbano

** The lack of a hyphen in the test set is only for the calculation of CER/WER.

After this error analysis, it became clear the need for more normalization rules for filled pauses representations so that the model accuracy increases.

5. Conclusions and Future Work

In this paper we presented and made publicly available a new dataset called CORAA v1, with 290.77 hours of validated pairs of audio-transcription, composed by public corpora in BP and TEDx Talks in European and Brazilian Portuguese.

Counting on the cooperation among research centers, universities, private companies and The São Paulo Research Foundation (FAPESP), we made publicly available this new and large dataset for training BP speech recognition models, closing the gap of the previous datasets, i.e., the lack of spontaneous and informal speech used in conversations, dialogues and interviews. Informed by the error analysis, we are normalizing filled pauses representations and performing a new validation over the test and development datasets, in order to increase future model accuracy.

As for future work, we plan to augment CORAA with new corpora from Tarsila Project[33] such as Museu da Pessoa[34] and NURC-SP[35]. We also plan to create an ASR Challenge including CORAA v1 to further develop research in ASR for the Portuguese language, in order to motivate young researchers in this exciting research area. Finally, we plan to refine the normalization rules of filled pauses and deliver a new version of CORAA dataset.

6. Acknowledgements

This research was funded by CEIA with support by the Goiás State Foundation (FAPEG grant #201910267000527)[36], Department of Higher Education of the Ministry of Education (SESU/MEC), Copel Holding S.A.[37], and Cyberlabs Group[38]. In addition, this research was carried out at the Center for Artificial Intelligence (C4AI-USP), with support by the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and by the IBM Corporation. Also, this study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. We also would like to thank Nvidia Corporation for the donation of Titan V GPU used in CORAA related projects. The coauthor Anderson da Silva Soares thanks to CNPq for Productivity Scholarship in Technological Development and Innovative Extension - number 308808/2020-7. Finally, the authors would like to thank all the members of the TARSILA project that contributed with discussions and insights regarding the compilation of CORAA v1 corpus.

[33] `https://sites.google.com/view/tarsila-c4ai`

[34] `https://museudapessoa.org/`

[35] `https://nurc.fflch.usp.br/`

[36] `http://centrodeia.org/`

[37] `https://www.copel.com`

[38] `https://cyberlabs.ai/`

Table 9: *Intervals of CER and frequencies of the different error types.*

Intervals	CER	Analysed Samples	Error Types (occurrences)
1 — 4,613	0	—	—
4,614 — 8,397	$0 < \text{CER} < 0.1$	110	1 (1), 2 (3), 4 (5), **5 (66)**, 6 (28), 8 (1), 9 (1), 10 (1), 11 (2)
8,398 — 10,724	$0.1 \leq \text{CER} < 0.2$	10	2 (3), **3 (8)**, 4 (1), 5 (1), 7 (1)
10,725 — 11,991	$0.2 \leq \text{CER} < 0.3$	10	2 (2), **3 (7)**, 8 (1)
11,992 — 12,666	$0.3 \leq \text{CER} < 0.4$	10	2 (2), **3 (14)**, 4 (1), 5 (3), 9 (1)
12,667 — 13,049	$0.4 \leq \text{CER} < 0.5$	25	2 (5), **3 (29)**, 4 (2), 5 (9), 6 (1), 8 (1)
13,050 — 13,336	$0.5 \leq \text{CER} < 0.6$	10	2 (2), **3 (6)**, 6 (1), 7 (1), 9 (1)
13,337 — 13,509	$0.6 \leq \text{CER} < 0.7$	10	2 (2), **3 (5)**, 5 (2), 6 (2), 8 (1)
13,510 — 13,932	$0.7 \leq \text{CER} \leq 12$	315	**3 (42)**, 6 (2), 8 (35), 9 (2)

7. References

[1] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," in *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 4218–4222. [Online]. Available: https://www.aclweb.org/anthology/2020.lrec-1.520

[2] C. Wang, J. Pino, A. Wu, and J. Gu, "CoVoST: A diverse multilingual speech-to-text translation corpus," in *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 4197–4203. [Online]. Available: https://www.aclweb.org/anthology/2020.lrec-1.517

[3] C. Wang, A. Wu, and J. Pino, "Covost 2: A massively multilingual speech-to-text translation corpus," 2020.

[4] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "Mls: A large-scale multilingual dataset for speech research," *Interspeech 2020*, Oct 2020. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-2826

[5] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.

[6] M. Zanon Boito, W. Havard, M. Garnerin, E. Le Ferrand, and L. Besacier, "Mass: A large and clean multilingual corpus of sentence-aligned spoken utterances extracted from the bible," in *Proceedings of The 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, May 2020, pp. 6486–6493. [Online]. Available: https://www.aclweb.org/anthology/2020.lrec-1.799

[7] F. Hernandez, V. Nguyen, S. Ghannay, N. A. Tomashenko, and Y. Estève, "TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation," in *Speech and Computer - 20th International Conference, SPECOM 2018, Leipzig, Germany, September 18-22, 2018, Proceedings*, ser. Lecture Notes in Computer Science, A. Karpov, O. Jokisch, and R. Potapova, Eds., vol. 11096. Springer, 2018, pp. 198–208. [Online]. Available: https://doi.org/10.1007/978-3-319-99579-3_21

[8] E. Salesky, M. Wiesner, J. Bremerman, R. Cattoni, M. Negri, M. Turchi, D. W. Oard, and M. Post, "The multilingual tedx corpus for speech recognition and translation," *CoRR*, vol. abs/2102.01757, 2021. [Online]. Available: https://arxiv.org/abs/2102.01757

[9] V. F. S. Alencar and A. Alcaim, "Lsf and lpc - derived features for large vocabulary distributed continuous speech recognition in brazilian portuguese," in *2008 42nd Asilomar Conference on Signals, Systems and Computers*, 2008, pp. 1237–1241.

[10] I. Macedo Quintanilha, S. Lima Netto, and L. Pereira Biscainho, "An open-source end-to-end asr system for brazilian portuguese using dnns built from newly assembled corpora," *Journal of Communication and Information Systems*, vol. 35, no. 1, pp. 230–242, Sep. 2020. [Online]. Available: https://jcis.sbrt.org.br/jcis/article/view/721

[11] H. Inaguma, K. Inoue, M. Mimura, and T. Kawahara, "Social Signal Detection in Spontaneous Dialogue Using Bidirectional LSTM-CTC," in *Proc. Interspeech 2017*, 2017, pp. 1691–1695.

[12] H. Fujimura, M. Nagao, and T. Masuko, "Simultaneous speech recognition and acoustic event detection using an lstm-ctc acoustic model and a wfst decoder," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5834–5838, 2018.

[13] T. Tanaka, R. Masumura, M. Ihori, A. Takashima, S. Orihashi, and N. Makishima, "End-to-End Rich Transcription-Style Automatic Speech Recognition with Semi-Supervised Learning," in *Proc. Interspeech 2021*, 2021, pp. 4458–4462.

[14] T. Baumann, C. Kennington, J. Hough, and D. Schlangen, "Recognising conversational speech: What an incremental ASR should do for a dialogue system and how to get there," in *Dialogues with Social Robots - Enablements, Analyses, and Evaluation, Seventh International Workshop on Spoken Dialogue Systems, IWSDS 2016, Saariselkä, Finland, January 13-16, 2016*, ser. Lecture Notes in Electrical Engineering, K. Jokinen and G. Wilcock, Eds., vol. 427. Springer, 2016, pp. 421–432. [Online]. Available: https://doi.org/10.1007/978-981-10-2585-3_35

[15] S. C. L. Gonçalves, "Projeto ALIP (amostra linguística do interior paulista) e banco de dados iboruna: 10 anos de contribuição com a descrição do português brasileiro," *Revista Estudos Linguísticos*, vol. 48, no. 1, pp. 276–297, dez. 2019.

[16] T. Raso and H. Mello, *C-oral - Brasil I: Corpus de Referência do Português Brasileiro Falado Informal*. Belo Horizonte, MG: Editora UFMG, 2012.

[17] M. Oliviera Jr., "Nurc digital um protocolo para a digitalização, anotação, arquivamento e disseminação do material do projeto da norma urbana linguística culta (nurc)," *CHIMERA: Revista de Corpus de Lenguas Romances y Estudios Lingüísticos*, vol. 3, no. 2, pp. 149–174, sep. 2016. [Online]. Available: https://revistas.uam.es/chimera/article/view/6519

[18] R. B. Mendes and L. Oushiro, "Mapping paulistano portuguese: the sp2010 project," in *Proceedings of the VIIth GSCP International Conference: Speech and Corpora*. Firenze, Italy: Fizenze University Press, 2012, pp. 459–463.

[19] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, É. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451.

[20] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[21] L. R. S. Gris, E. Casanova, F. S. de Oliveira, A. da Silva Soares, and A. C. Junior, "Brazilian portuguese speech recognition using wav2vec 2.0," 2021.

[22] S. C. L. Gonçalves, "Banco de dados Iboruna: amostras eletrônicas do português falado no interior paulista," https://www.alip.ibilce.unesp.br/, 2021, accessed: 2021-07-1.

[23] T. Raso, H. Mello, and M. Mittmann, "O projeto c-oral-brasil," *CHIMERA: Revista de Corpus de Lenguas Romances y Estudios Lingüísticos*, vol. 1, p. 31–67, feb. 2015. [Online]. Available: https://revistas.uam.es/chimera/article/view/249

[24] T. Raso, H. Mello, and M. M. Mittmann, "The C-ORAL-BRASIL I: Reference corpus for spoken Brazilian Portuguese," in *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*. Istanbul, Turkey: European Language Resources Association (ELRA), May 2012, pp. 106–113. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2012/pdf/624_Paper.pdf

[25] M. M. Emanuela Cresti, Lorenzo Gregori and A. Panunzi, "The language into act theory: A pragmatic approach to speech in real-life," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, H. Koiso and P. Paggio, Eds. Paris, France: European Language Resources Association (ELRA), may 2018.

[26] E. Bick, *The Parsing System "Palavras". Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Århus: University of Arhus, 2000.

[27] Projeto SP2010, "Projeto SP2010: Amostra da fala paulistana," https://projetosp2010.fflch.usp.br/corpus, 2021, accessed: 2021-07-11.

[28] D. Preti, "Normas para transcrição dos exemplos," in *Análise de Textos Orais*, 4th ed., ser. Série Projetos Paralelos, D. Preti, Ed. Humanitas Publicações - FFLCH/USP, Junho 1999, vol. 1, pp. 11–12.

[29] T. Raso and H. Mello, "Parâmetros de compilação de um corpus oral: o caso do c-oral-brasi," *Veredas*, vol. 13, p. 20–35, 2009. [Online]. Available: https://periodicos.ufjf.br/index.php/veredas/article/view/25149

[30] J. R. Landis and G. G. Koch, "The measurement of observer agreement for categorical data," *Biometrics*, vol. 33, no. 1, pp. 159–174, 1977. [Online]. Available: http://www.jstor.org/stable/2529310

[31] A. K. Sheshadri, A. Rao Vijjini, and S. Kharbanda, "WER-BERT: Automatic WER estimation with BERT in a balanced ordinal classification paradigm," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, Apr. 2021, pp. 3661–3672. [Online]. Available: https://aclanthology.org/2021.eacl-main.320

[32] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, and A. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.6

[33] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7

[34] M. da Mota, A. H. Moussatchè, C. R. de Castro, M. L. S. de Moura, and T. D'Angelis, "Erros de escrita no contexto: uma análise na abordagem do processamento da informação," *Psicologia: Reflexão e Crítica [online]*, vol. 13, no. 1, 2000.

CAPÍTULO 9

CONCLUSÕES

O objetivo geral deste trabalho consistiu na investigação de uma abordagem para o desenvolvimento de um método *zero-shot multi-speaker TTS* para idiomas que possuem *datasets* com baixa quantidade de locutores disponíveis. Deste modo, buscando tornar mais viável a utilização de síntese de fala aplicada à tarefa de ASR em idiomas com poucos recursos disponíveis. O objetivo geral dessa pesquisa foi atingido após uma série de estudos, culminando com o artigo apresentado no Capítulo 6, no qual é proposto um modelo *zero-shot multi-speaker TTS* que permite realizar a tarefa em um idioma alvo utilizando apenas um locutor nesse idioma. Além disso, a aplicação do método de aumento de dados proposto na tarefa de ASR, foi bem sucedida e apresentada no Capítulo 7.

Os resultados obtidos permitem traçar conclusões sobre a questão de pesquisa central da tese "**dado que sistemas *zero-shot multi-speaker TTS* exigem *datasets* com um grande número de locutores para sua convergência, é possível superar essa limitação e obter um sistema *zero-shot multi-speaker TTS* em idiomas para os quais o número de locutores disponíveis tende a um** ?". Com o desenvolvimento deste trabalho, podemos afirmar que é possível superar a limitação do número de locutores disponíveis em um idioma e desenvolver um sistema *zero-shot multi-speaker TTS* em idiomas que possuem *datasets* com poucos locutores disponíveis.

Em particular, o Capítulo 6 apresentou resultados muito promissores com o treinamento do modelo de síntese de fala proposto com um único locutor no idioma português. Além disso, o Capítulo 7 apresentou a aplicação do modelo de síntese de fala proposto, treinado com apenas um locutor nos idiomas alvos, no aumento de dados para o treinamento de um modelo ASR. O método proposto alcançou resultados comparáveis ao estado da arte no idioma inglês, deste modo, mostrando que o método proposto é bom o suficiente para ser empregado no aumento de dados para tarefa de ASR.

A hipótese da pesquisa era a de que um modelo *flow-based*, como o GlowTTS, adaptado para treinamento *zero-shot multi-speaker* poderia alcançar a convergência com uma menor quantidade de locutores. Essa hipótese foi confirmada no Capítulo 5, no qual o modelo *flow-based* SC-GlowTTS alcançou resultados próximos ao estado da arte em *zero-shot multi-speaker TTS* no *dataset* VCTK utilizando apenas 11 locutores durante o treinamento. Além disso, nossa hipótese incluía o estudo do treinamento de um modelo usufruindo da quantidade de locutores presentes em múltiplos idiomas, desta forma diminuindo a quantidade necessária de locutores para treinamento no idioma alvo. Isso foi confirmado no Capítulo 6, no qual o modelo multilíngue YourTTS alcançou resultados promissores em um idioma alvo utilizando apenas um locutor no idioma alvo. Além disso, o modelo foi capaz de reproduzir vozes femininas mesmo sem ter sido treinado com vozes femininas no idioma alvo.

Em conclusão, os métodos desenvolvidos nessa pesquisa de doutorado viabilizam a aplicação de síntese de fala na tarefa de ASR em idiomas para os quais apenas *datasets* compostos por um único locutor estão disponíveis. Deste modo, essa nova tecnologia não está restrita apenas para idiomas com muitos recursos disponíveis, podendo ser aplicada para qualquer idioma que possua um *dataset* composto por pelo menos um locutor com o número de horas suficiente para a convergência do modelo TTS[1].

Por fim, além das contribuições para a área de reconhecimento automático de fala, os métodos desenvolvidos nessa pesquisa de doutorado permitem a exploração de potenciais aplicações de modelos *zero-shot multi-speaker TTS* e *cross-lingual zero-shot voice conversion*, que antes eram limitados apenas a idiomas com muitos recursos disponíveis. As aplicações são, por exemplo, (i) sistemas de síntese de fala para locutores que perderam a capacidade de fala ou tiveram a diminuição da capacidade de fala, desenvolvidos com a utilização de apenas alguns segundos de fala, (ii) dublagem de filmes e séries utilizando a voz original do ator em outros idiomas, e (iii) geração de novas vozes para sistemas de autoatendimento.

9.1 Detalhamento das Contribuições

Inicialmente, visando contribuir com a questão de falta de dados para síntese de fala no português brasileiro, foi proposto e disponibilizado publicamente o *dataset TTS-Portuguese Corpus*, detalhado no Capítulo 4, que é o primeiro *dataset* disponível publicamente para treinamento de modelos profundos de síntese de fala em português brasileiro. Modelos treinados no *dataset* alcançaram resultados comparáveis ao idioma inglês e modelos treinados com *datasets* não públicos no português.

Seguindo a linha de avanço dos modelos *flow-based*, foi proposto o *Speaker Conditional GlowTTS (SC-GlowTTS)*, detalhado no Capítulo 5, que é um modelo *zero-shot multi-speaker TTS* com resultados no estado da arte no idioma inglês. Além disso, o SC-GlowTTS foi integrado com o *vocoder* HiFi-GAN (KONG; KIM; BAE, 2020), mostrando que o ajuste do *vocoder*

[1] No Capítulo 7, experimentos mostraram que 8,38 horas e 14,94 horas foram suficientes para o português e o russo, respectivamente.

nos espectrogramas preditos do conjunto de treinamento pelo modelo TTS melhora de forma significativa a qualidade e a similaridade da fala sintetizada para novos locutores, diminuindo assim a lacuna de similaridade entre locutores vistos e não vistos no treinamento do modelo. Além da síntese de fala na voz de locutores não vistos no treinamento, o SC-GlowTTS permite a conversão de voz para locutores não vistos no treinamento (*zero-shot voice conversion*). Por fim, o modelo alcança resultados próximos ao estado da arte, utilizando apenas 11 locutores no treinamento.

Apesar dos resultados promissores do modelo SC-GlowTTS, a lacuna de similaridade entre locutores vistos e não vistos durante o treinamento ainda é uma questão de pesquisa aberta. De acordo com Tan *et al.* (2021) a qualidade dos modelos *zero-shot multi-speaker TTS* atuais não é suficientemente boa, especialmente para locutores com características de fala muito diferentes das vistas no treinamento. Além disso, os modelos *zero-shot multi-speaker TTS* ainda requeriam um grande número de locutores para o treinamento, dificultando a obtenção de modelos de alta qualidade em línguas com poucos recursos. Embora o modelo SC-GlowTTS tenha alcançado resultados promissores com apenas 11 locutores do *dataset* VCTK, geralmente, limitar o número e a variedade de locutores no treinamento dificulta ainda mais a generalização do modelo para vozes com características muito diferentes das vistas durante o treinamento do modelo.

Buscando mitigar esses problemas, foi proposto o YourTTS, detalhado no Capítulo 6. YourTTS é um modelo *zero-shot multi-speaker TTS* multilíngue, baseado nos modelos VITS e SC-GlowTTS, que alcança resultados estado da arte em *zero-shot multi-speaker TTS* e resultados comparáveis ao estado da arte em *zero-shot voice conversion* no idioma inglês (dataset VCTK). Além disso, o modelo alcançou resultados promissores em *zero-shot voice conversion* e *zero-shot multi-speaker TTS* utilizando apenas um único locutor em um idioma alvo. O modelo ainda foi capaz de, por exemplo, produzir voz feminina no idioma alvo sem nunca ter sido treinado com vozes femininas nesse idioma. Por fim, para locutores que possuem características de fala muito diferentes das vistas no treinamento o modelo YourTTS pode ser ajustado utilizando apenas 1 minuto de fala desses locutores e alcançar resultados no estado da arte. Portanto, o modelo YourTTS torna possível a realização de *zero-shot multi-speaker TTS* e *zero-shot voice conversion* em um idioma alvo utilizando, durante o treinamento, apenas um locutor nesse idioma, deste modo, sendo um caminho viável para o desenvolvimento desses sistemas em idiomas com poucos recursos disponíveis. Além disso, apesar de não ter sido explorado no artigo o modelo permite *code-switching*.

No Capitulo 7, explorou-se o uso do modelo YourTTS no aumento de dados para o treinamento de modelos ASR. Foi possível melhorar o desempenho do modelo ASR de forma comparável ao estado da arte no idioma inglês. Além disso, mostrou-se a viabilidade da utilização de síntese de fala como aumento de dados em um idioma alvo, a partir do uso de um único locutor nesse idioma. Isso permite mitigar limitações reportadas por Laptev *et al.* (2020), abrindo novas possibilidades para o treinamento de modelos ASR em idiomas com poucos recursos disponíveis.

Por fim, o Capítulo 8 apresentou o trabalho de adaptação de um grande *dataset* para a tarefa de ASR, composto por 290 horas de fala espontânea (o primeiro deste tipo), principalmente, no português brasileiro. Além disso, o artigo disponibiliza publicamente um modelo de ASR treinado nesse grande dataset.

Essa pesquisa de doutorado também traz contribuições para a área de processamento de fala não relacionadas com o tema da tese — síntese de fala aplicada ao reconhecimento automático de fala. Inicialmente, trazendo contribuições para a tarefa de detecção de limite de sentenças em fala comprometida (Apêndice A.1). Além disso, devido à pandemia do COVID-19, e da necessidade de identificação dessa doença de forma rápida e com o menor contato físico possível, esse trabalho buscou contribuir com a identificação de insuficiência respiratória em pacientes com COVID-19 (Apêndice A.2) e com a identificação de COVID-19 (Apêndice A.3) através da fala e tosse. Por fim, foi proposto o Speech2Phone, detalhado no Apêndice A.4, que é um método eficiente para treinar modelos de verificação de locutores. O Speech2Phone alcançou resultados compatíveis com o estado da arte utilizando 500x menos dados de treinamento.

9.2 Trabalhos Futuros

Em trabalhos futuros, pretende-se melhorar o preditor de duração do modelo YourTTS, bem como realizar o treinamento com mais idiomas, deste modo facilitando a convergência do modelo em novos idiomas, especialmente em idiomas com poucos recursos disponíveis como por exemplo idiomas indígenas. Além disso, pretende-se explorar o uso do extrator de características do modelo Wav2vec 2.0 como um discriminador durante o treinamento do modelo YourTTS, deste modo forçando o modelo YourTTS a sintetizar fala mais próxima da fala humana para o modelo Wav2vec 2.0 e possivelmente diminuir a lacuna de desempenho entre modelos treinados com fala humana e fala sintetizada. Por fim, pretende-se investigar métodos para a identificação de *deep fake* em áudio, buscando fornecer meios eficientes de detectar o uso indevido dos modelos de *zero-shot multi-speaker TTS* propostos nesta tese de doutorado.

9.3 Publicações: artigos publicados e submetidos

Foram escritos 13 artigos durante essa pesquisa, 11 já publicados, dois em processo de revisão em revistas, que foram incluídos nos Capítulos 7 e 8.

A Tabela 9 apresenta em ordem cronológica todos os artigos publicados e submetidos no decorrer dessa pesquisa, com indicação dos trabalhos diretamente relacionados com a tese da pesquisa.

Tabela 9 – Lista de artigos publicados e submetidos durante esta pesquisa, em ordem cronológica.

Publicações	**Relacionada com a Tese**
CABEZUDO, M. A. S.; INÁCIO, M.; RODRIGUES, A. C.; CASANOVA, E.; DE SOUSA, R. F.. **NILC at ASSIN 2: Exploring Multilingual Approaches**. In: ASSIN@STIL. 2019. p. 49-58.	Não
CABEZUDO, M. A. S.; INÁCIO, M.; RODRIGUES, A. C.; CASANOVA, E.; DE SOUSA, R. F.. **Natural Language Inference for Portuguese Using BERT and Multilingual Information**. In: Proceedings of The International Conference on the Computational Processing of Portuguese (PROPOR). Springer, Cham, 2020. p. 346-356.	Não
CASANOVA, E.; TREVISO, M.; HÜBNER, L.; ALUÍSIO, S.. **Evaluating Sentence Segmentation in Different Datasets of Neuropsychological Language Tests in Brazilian Portuguese**. In: Proceedings of The 12th Language Resources and Evaluation Conference (LREC 2020). Marseille, France: European Language Resources Association (ELRA), 2020. p. 2605-2614.	Não
GRIS, L. R. S.; CASANOVA, E.; DE OLIVEIRA, F. S.; SOARES, A. S.; CANDIDO Jr, A.. **Desenvolvimento de um modelo de reconhecimento de voz para o Português Brasileiro com poucos dados utilizando o Wav2vec 2.0**. In Anais do XV Brazilian e-Science Workshop. SBC., 2021. p. 129-136.	Não
CASANOVA, E. ; GRIS, L. ; CAMARGO, A. ; SILVA, D. ; GAZZOLA, M.; SABINO, E.; LEVIN, A.; CANDIDO JR, A. ; ALUISIO, S.; FINGER, M.. **Deep learning against covid-19: Respiratory insufficiency detection in brazilian portuguese speech**. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. ACL, Aug. 2021.	Não
CASANOVA, E.; CANDIDO JR, A.; FERNANDES JR, R. C.; Finger, M.; GRIS, L.; PONTI, M. A.. **Transfer Learning and Data Augmentation Techniques to the COVID-19 Identification Tasks in ComParE 2021**. In: Proceedings of INTERSPEECH. ISCA, Aug. 2021.	Não
CASANOVA, E.; SHULBY, C.; GÖLGE, E.; MÜLLER, N. M.; DE OLIVEIRA, F. S.; CANDIDO Jr, A. ; SOARES, A. S.; ALUISIO, S.; PONTI, M. A.. **SC-GlowTTS: an Efficient Zero-Shot Multi-Speaker Text-To-Speech Model**. In: Proceedings of INTERSPEECH. ISCA, Aug. 2021.	Sim
LEAL, S.; CASANOVA, E.; PAETZOLD, G.; ALUISIO, S.. **Evaluating Semantic Similarity Methods to Build Semantic Predictability Norms of Reading Data**. In: Proceedings of the 24th International Conference on Text, Speech and Dialogue, TSD 2021. ISCA, Sept. 2021.	Não

CASANOVA, E.; CANDIDO JR, A.; SHULBY, C.; DE OLIVEIRA, F. S., GRIS, L. R. S., DA SILVA, H. P.; PONTI, M. A. **Speech2Phone: A Novel and Efficient Method for Training Speaker Recognition Models**. In: Brazilian Conference on Intelligent Systems. Springer, Cham, Dec. 2021. p 572-585.	Não
CASANOVA, E.; CANDIDO JR, A.; SHULBY, C.; DE OLIVEIRA, F. S.; TEIXEIRA, J. P.; PONTI, M. A.; ALUISIO, S.. **TTS-Portuguese Corpus: a corpus for speech synthesis in Brazilian Portuguese**. In: Language Resources and Evaluation (LREV). Springer, 2022.	Sim
CASANOVA, E.; WEBER, J.; SHULBY, C.; JUNIOR, A. C.; GÖLGE, E.; PONTI, M. A.. **YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone**. In: Proceedings of International Conference on Machine Learning (ICML). PMLR, 2022.	Sim
CANDIDO JR, A.; CASANOVA, E.; SOARES, A.; DE OLIVEIRA, F. S.; OLIVEIRA, L.; JUNIOR, R. C. F.; ... ; ALUISIO, S.. **CORAA: a large corpus of spontaneous and prepared speech manually validated for speech recognition in Brazilian Portuguese**, In: Language Resources and Evaluation (LREV). Springer, 2022, accepted for publication with revisions.	Sim
CASANOVA, E.; SHULBY, C.; CANDIDO JR, A.; ALUÍSIO, S.; PONTI, M. A.. **A single speaker is almost all you need for automatic speech recognition**. In: IEEE Signal Processing Letters. IEEE, 2022, under review.	Sim

REFERÊNCIAS

ABDELHAMED, A.; BRUBAKER, M. A.; BROWN, M. S. Noise flow: Noise modeling with conditional normalizing flows. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2019. p. 3165–3173. Citado nas páginas 52 e 53.

ABOUELHASAN, N.; ELBORAEE, T.; MOHAMED, H.; ADEL, N.; EID, M. M. Survey on automatic speech recognition. **Journal of Computer Science and Information Systems**, v. 11, n. 2 June 2020, 2020. Citado na página 22.

ALI, A.; RENALS, S. Word error rate estimation for speech recognition: e-wer. In: **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**. [S.l.: s.n.], 2018. p. 20–24. Citado na página 56.

AMODEI, D.; ANANTHANARAYANAN, S.; ANUBHAI, R.; BAI, J.; BATTENBERG, E.; CASE, C.; CASPER, J.; CATANZARO, B.; CHENG, Q.; CHEN, G. *et al.* Deep speech 2: End-to-end speech recognition in english and mandarin. In: **International Conference on Machine Learning**. [S.l.: s.n.], 2016. p. 173–182. Citado nas páginas 21, 22 e 36.

ANTHIMOPOULOS, M.; CHRISTODOULIDIS, S.; EBNER, L.; CHRISTE, A.; MOUGIAKAKOU, S. Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. **IEEE transactions on medical imaging**, IEEE, v. 35, n. 5, p. 1207–1216, 2016. Citado na página 35.

ARDILA, R.; BRANSON, M.; DAVIS, K.; HENRETTY, M.; KOHLER, M.; MEYER, J.; MORAIS, R.; SAUNDERS, L.; TYERS, F. M.; WEBER, G. Common voice: A massively-multilingual speech corpus. In: . [S.l.: s.n.], 2019. Citado nas páginas 62 e 119.

ARIK, S.; CHEN, J.; PENG, K.; PING, W.; ZHOU, Y. Neural voice cloning with a few samples. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2018. p. 10019–10029. Citado na página 24.

ARIK, S. Ö.; CHRZANOWSKI, M.; COATES, A.; DIAMOS, G.; GIBIANSKY, A.; KANG, Y.; LI, X.; MILLER, J.; NG, A.; RAIMAN, J. *et al.* Deep voice: Real-time neural text-to-speech. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2017. p. 195–204. Citado na página 23.

BA, J. L.; KIROS, J. R.; HINTON, G. E. Layer normalization. **arXiv preprint arXiv:1607.06450**, 2016. Citado nas páginas 41 e 48.

BAEVSKI, A.; HSU, W.-N.; XU, Q.; BABU, A.; GU, J.; AULI, M. Data2vec: A general framework for self-supervised learning in speech, vision and language. **arXiv preprint arXiv:2202.03555**, 2022. Citado nas páginas 21, 23 e 55.

BAEVSKI, A.; ZHOU, Y.; MOHAMED, A.; AULI, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. **Advances in Neural Information Processing Systems**, v. 33, 2020. Citado nas páginas 21, 22, 23, 25, 40 e 55.

BAHDANAU, D.; CHO, K. H.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. In: **3rd International Conference on Learning Representations, ICLR 2015**. [S.l.: s.n.], 2015. Citado nas páginas 23 e 46.

BAHDANAU, D.; CHOROWSKI, J.; SERDYUK, D.; BRAKEL, P.; BENGIO, Y. End-to-end attention-based large vocabulary speech recognition. In: IEEE. **2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2016. p. 4945–4949. Citado na página 23.

BAKER, J.; DENG, J. L.; GLASS, J.; KHUDANPUR, S.; LEE, S. Chin-hui; MORGAN, N.; O'SHAUGHNESSY, D. Research developments and directions in speech recognition and understanding, part 1. In: INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. [S.l.], 2009. Citado na página 22.

BENESTY, J.; SONDHI, M. M.; HUANG, Y. **Springer handbook of speech processing**. [S.l.]: Springer, 2007. Citado na página 21.

BIŃKOWSKI, M.; DONAHUE, J.; DIELEMAN, S.; CLARK, A.; ELSEN, E.; CASAGRANDE, N.; COBO, L. C.; SIMONYAN, K. High fidelity speech synthesis with adversarial networks. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2019. Citado na página 23.

BJORCK, J.; GOMES, C.; SELMAN, B.; WEINBERGER, K. Q. Understanding batch normalization. In: **Proceedings of the 32nd International Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2018. p. 7705–7716. Citado na página 41.

BOSMAN, P. A.; THIERENS, D. Negative log-likelihood and statistical hypothesis testing as the basis of model selection in ideas. 2000. Citado na página 53.

CAI, W.; CHEN, J.; LI, M. Exploring the encoding layer and loss function in end-to-end speaker and language recognition system. In: **Proc. Odyssey 2018 The Speaker and Language Recognition Workshop**. [S.l.: s.n.], 2018. p. 74–81. Citado na página 25.

CAO, Y.; WU, X.; LIU, S.; YU, J.; LI, X.; WU, Z.; LIU, X.; MENG, H. End-to-end code-switched tts with mix of monolingual recordings. In: IEEE. **ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2019. p. 6935–6939. Citado nas páginas 26, 60 e 61.

CASANOVA, E. **Síntese de voz aplicada ao português brasileiro usando aprendizado profundo**. [S.l.]: Universidade Tecnológica Federal do Paraná, 2019. Citado na página 31.

CAVALLARI, G. B.; RIBEIRO, L. S.; PONTI, M. A. Unsupervised representation learning using convolutional and stacked auto-encoders: a domain and cross-domain feature space analysis. In: IEEE. **2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.], 2018. p. 440–446. Citado na página 49.

CHAN, W.; JAITLY, N.; LE, Q.; VINYALS, O. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In: IEEE. **2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2016. p. 4960–4964. Citado nas páginas 21, 22 e 23.

CHENG, J.-M.; WANG, H.-C. A method of estimating the equal error rate for automatic speaker verification. In: IEEE. **2004 International Symposium on Chinese Spoken Language Processing**. [S.l.], 2004. p. 285–288. Citado na página 56.

CHIU, C.-C.; SAINATH, T. N.; WU, Y.; PRABHAVALKAR, R.; NGUYEN, P.; CHEN, Z.; KANNAN, A.; WEISS, R. J.; RAO, K.; GONINA, E. *et al.* State-of-the-art speech recognition with sequence-to-sequence models. In: IEEE. **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2018. p. 4774–4778. Citado nas páginas 23 e 58.

CHO, K.; MERRIENBOER, B. van; GULCEHRE, C.; BOUGARES, F.; SCHWENK, H.; BENGIO, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: **Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)**. [S.l.: s.n.], 2014. Citado na página 44.

CHOI, S.; HAN, S.; KIM, D.; HA, S. Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding. **Proc. Interspeech 2020**, p. 2007–2011, 2020. Citado na página 25.

CHUNG, J.; GULCEHRE, C.; CHO, K.; BENGIO, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In: **NIPS 2014 Workshop on Deep Learning, December 2014**. [S.l.: s.n.], 2014. Citado na página 39.

CHUNG, J. S.; HUH, J.; MUN, S.; LEE, M.; HEO, H.-S.; CHOE, S.; HAM, C.; JUNG, S.; LEE, B.-J.; HAN, I. In defence of metric learning for speaker recognition. **Proc. Interspeech 2020**, p. 2977–2981, 2020. Citado na página 36.

CHUNG, J. S.; HUH, J.; MUN, S.; LEE, M.; HEO, H. S.; CHOE, S.; HAM, C.; JUNG, S.; LEE, B.-J.; HAN, I. In defence of metric learning for speaker recognition. In: **Interspeech**. [S.l.: s.n.], 2020. Citado nas páginas 40 e 54.

CHUNG, J. S.; NAGRANI, A.; ZISSERMAN, A. Voxceleb2: Deep speaker recognition. **Proc. Interspeech 2018**, p. 1086–1090, 2018. Citado na página 40.

COLLOBERT, R.; PUHRSCH, C.; SYNNAEVE, G. Wav2letter: an end-to-end convnet-based speech recognition system. **arXiv preprint arXiv:1609.03193**, 2016. Citado nas páginas 22, 55 e 58.

COLOMBO, P.; CHAPUIS, E.; MANICA, M.; VIGNON, E.; VARNI, G.; CLAVEL, C. Guiding attention in sequence-to-sequence models for dialogue act prediction. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2020. v. 34, n. 05, p. 7594–7601. Citado na página 44.

COOPER, E.; LAI, C.-I.; YASUDA, Y.; FANG, F.; WANG, X.; CHEN, N.; YAMAGISHI, J. Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings. In: IEEE. **ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2020. p. 6184–6188. Citado na página 25.

DANG, L. M.; HASSAN, S. I.; IM, S.; LEE, J.; LEE, S.; MOON, H. Deep learning based computer generated face identification using convolutional neural network. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 8, n. 12, p. 2610, 2018. Citado na página 36.

DAVIS, K. H.; BIDDULPH, R.; BALASHEK, S. Automatic recognition of spoken digits. **The Journal of the Acoustical Society of America**, v. 24, n. 6, p. 637–642, 1952. Disponível em: <https://doi.org/10.1121/1.1906946>. Citado na página 22.

DAVIS, S. B.; MERMELSTEIN, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. In: **Readings in speech recognition**. [S.l.]: Elsevier, 1990. p. 65–74. Citado na página 54.

DAYAN, P.; HINTON, G. E.; NEAL, R. M.; ZEMEL, R. S. The helmholtz machine. **Neural computation**, MIT Press, v. 7, n. 5, p. 889–904, 1995. Citado na página 49.

DEMPSEY, P. The teardown: Google home personal assistant. **Engineering & Technology**, IET, v. 12, n. 3, p. 80–81, 2017. Citado na página 22.

DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: IEEE. **Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on**. [S.l.], 2009. p. 248–255. Citado na página 40.

DENG, L.; LI, X. Machine learning paradigms for speech recognition: An overview. **IEEE Transactions on Audio, Speech, and Language Processing**, IEEE, v. 21, n. 5, p. 1060–1089, 2013. Citado na página 26.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**. [S.l.: s.n.], 2019. p. 4171–4186. Citado na página 48.

DINH, L.; KRUEGER, D.; BENGIO, Y. Nice: Non-linear independent components estimation. **arXiv preprint arXiv:1410.8516**, 2014. Citado nas páginas 52 e 53.

DINH, L.; SOHL-DICKSTEIN, J.; BENGIO, S. Density estimation using real nvp. **arXiv preprint arXiv:1605.08803**, 2016. Citado na página 53.

DONAHUE, J.; DIELEMAN, S.; BINKOWSKI, M.; ELSEN, E.; SIMONYAN, K. End-to-end adversarial text-to-speech. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2021. Citado nas páginas 23 e 24.

DOSOVITSKIY, A.; BEYER, L.; KOLESNIKOV, A.; WEISSENBORN, D.; ZHAI, X.; UNTERTHINER, T.; DEHGHANI, M.; MINDERER, M.; HEIGOLD, G.; GELLY, S. *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2020. Citado na página 48.

DURKAN, C.; BEKASOV, A.; MURRAY, I.; PAPAMAKARIOS, G. Neural spline flows. In: **Advances in Neural Information Processing Systems**. [S.l.: s.n.], 2019. p. 7511–7522. Citado na página 24.

ELIZABETH, S.; MATTHEW, W.; JACOB, B.; CATTONI, R.; NEGRI, M.; TURCHI, M.; OARD, D. W.; MATT, P. The multilingual tedx corpus for speech recognition and translation. In: **Interspeech 2021**. [S.l.: s.n.], 2021. p. 3655–3659. Citado na página 119.

ERVEN, T. V.; HARREMOS, P. Rényi divergence and kullback-leibler divergence. **IEEE Transactions on Information Theory**, IEEE, v. 60, n. 7, p. 3797–3820, 2014. Citado na página 50.

FARFADE, S. S.; SABERIAN, M. J.; LI, L.-J. Multi-view face detection using deep convolutional neural networks. In: ACM. **Proceedings of the 5th ACM on International Conference on Multimedia Retrieval**. [S.l.], 2015. p. 643–650. Citado na página 36.

FIRAT, O.; CHO, K.; BENGIO, Y. Multi-way, multilingual neural machine translation with a shared attention mechanism. In: **Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**. [S.l.: s.n.], 2016. p. 866–875. Citado na página 38.

FREY, B. J.; BRENDAN, J. F.; FREY, B. J. **Graphical models for machine learning and digital communication**. [S.l.]: MIT press, 1998. Citado na página 52.

GALES, M.; YOUNG, S. *et al.* The application of hidden markov models in speech recognition. **Foundations and Trends® in Signal Processing**, Now Publishers, Inc., v. 1, n. 3, p. 195–304, 2008. Citado na página 22.

GEHRING, J.; AULI, M.; GRANGIER, D.; YARATS, D.; DAUPHIN, Y. N. Convolutional sequence to sequence learning. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2017. p. 1243–1252. Citado na página 23.

GIBIANSKY, A.; ARIK, S.; DIAMOS, G.; MILLER, J.; PENG, K.; PING, W.; RAIMAN, J.; ZHOU, Y. Deep voice 2: Multi-speaker neural text-to-speech. **Advances in neural information processing systems**, v. 30, 2017. Citado na página 23.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: **Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics**. [S.l.: s.n.], 2011. p. 315–323. Citado na página 34.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [S.l.]: MIT Press, 2016. Http://www.deeplearningbook.org. Citado na página 49.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A.; BENGIO, Y. **Deep learning**. [S.l.]: MIT press Cambridge, 2016. v. 1. Citado nas páginas 21, 31, 34, 35, 36, 37, 39, 41, 48 e 52.

GOODFELLOW, I.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. Generative adversarial networks. **Communications of the ACM**, ACM New York, NY, USA, v. 63, n. 11, p. 139–144, 2020. Citado na página 52.

GOODFELLOW, I. J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A. C.; BENGIO, Y. Generative adversarial nets. In: **NIPS**. [S.l.: s.n.], 2014. Citado na página 52.

GORDILLO, C. D. A. **Reconhecimento de Voz Contınua Combinando os Atributos MFCC e PNCC com Métodos de Robustez SS, WD, MAP e FRN**. Tese (Doutorado) — PUC-Rio, 2013. Citado na página 55.

GRAVES, A. Generating sequences with recurrent neural networks. **arXiv preprint arXiv:1308.0850**, 2013. Citado na página 44.

GRAVES, A.; FERNÁNDEZ, S.; GOMEZ, F.; SCHMIDHUBER, J. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: ACM. **Proceedings of the 23rd international conference on Machine learning**. [S.l.], 2006. p. 369–376. Citado na página 23.

GRAVES, A.; JAITLY, N.; MOHAMED, A.-r. Hybrid speech recognition with deep bidirectional lstm. In: IEEE. **2013 IEEE workshop on automatic speech recognition and understanding**. [S.l.], 2013. p. 273–278. Citado na página 38.

GRAVES, A.; MOHAMED, A.-r.; HINTON, G. Speech recognition with deep recurrent neural networks. In: IEEE. **2013 IEEE international conference on acoustics, speech and signal processing**. [S.l.], 2013. p. 6645–6649. Citado na página 23.

GREFF, K.; SRIVASTAVA, R. K.; SCHMIDHUBER, J. Highway and residual networks learn unrolled iterative estimation. **arXiv preprint arXiv:1612.07771**, 2016. Citado nas páginas 42 e 43.

GRUBER, T. R. Siri, a virtual personal assistant-bringing intelligence to the interface. In: **Semantic Technologies Conference**. [S.l.: s.n.], 2009. Citado na página 22.

GULATI, A.; QIN, J.; CHIU, C.-C.; PARMAR, N.; ZHANG, Y.; YU, J.; HAN, W.; WANG, S.; ZHANG, Z.; WU, Y.; AL. et. Conformer: Convolution-augmented transformer for speech recognition. **Interspeech 2020**, ISCA, Oct 2020. Disponível em: <http://dx.doi.org/10.21437/interspeech.2020-3015>. Citado nas páginas 21 e 23.

GUO, J. BackPropagation Through Time. **Manuscript**, n. 1, p. 1–6, 2013. Citado na página 40.

GUYON, I.; MAKHOUL, J.; SCHWARTZ, R.; VAPNIK, V. What size test set gives good error rate estimates? **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 20, n. 1, p. 52–64, 1998. Citado na página 56.

HANNUN, A.; CASE, C.; CASPER, J.; CATANZARO, B.; DIAMOS, G.; ELSEN, E.; PRENGER, R.; SATHEESH, S.; SENGUPTA, S.; COATES, A. *et al.* Deep speech: Scaling up end-to-end speech recognition. **arXiv preprint arXiv:1412.5567**, 2014. Citado nas páginas 21, 22 e 55.

HAYKIN, S. **Neural networks and learning machines, 3/E**. [S.l.]: Pearson Education India, 2010. Citado nas páginas 33, 34 e 36.

HAYKIN, S. S.; HAYKIN, S. S.; HAYKIN, S. S.; HAYKIN, S. S. **Neural networks and learning machines**. [S.l.]: Pearson Upper Saddle River, NJ, USA:, 2009. v. 3. Citado nas páginas 32 e 38.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**. [S.l.: s.n.], 2016. p. 770–778. Citado nas páginas 36 e 42.

HIJAZI, S.; KUMAR, R.; ROWEN, C. Using convolutional neural networks for image recognition. **Cadence Design Systems Inc.: San Jose, CA, USA**, 2015. Citado na página 36.

HINTON, G. E.; OSINDERO, S.; TEH, Y.-W. A fast learning algorithm for deep belief nets. **Neural computation**, MIT Press, v. 18, n. 7, p. 1527–1554, 2006. Citado na página 40.

HINTON, G. E.; SALAKHUTDINOV, R. R. Reducing the dimensionality of data with neural networks. **science**, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006. Citado na página 48.

HO, J.; CHEN, X.; SRINIVAS, A.; DUAN, Y.; ABBEEL, P. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2019. p. 2722–2730. Citado na página 52.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, MIT Press, v. 9, n. 8, p. 1735–1780, 1997. Citado na página 39.

HOLTER, T.; HARBORG, E.; JOHNSEN, M. H.; SVENDSEN, T. Asr-based subtitling of live tv-programs for the hearing impaired. In: **Sixth International Conference on Spoken Language Processing**. [S.l.: s.n.], 2000. Citado na página 22.

HOOGEBOOM, E.; BERG, R. V. D.; WELLING, M. Emerging convolutions for generative normalizing flows. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2019. p. 2771–2780. Citado na página 24.

HORNIK, K. Approximation capabilities of multilayer feedforward networks. **Neural networks**, Elsevier, v. 4, n. 2, p. 251–257, 1991. Citado na página 33.

HSU, W.-N.; TSAI, Y.-H. H.; BOLTE, B.; SALAKHUTDINOV, R.; MOHAMED, A. Hubert: How much can a bad teacher benefit asr pre-training? In: IEEE. **ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2021. p. 6533–6537. Citado nas páginas 21, 23, 40 e 55.

HSU, W.-N.; ZHANG, Y.; WEISS, R. J.; ZEN, H.; WU, Y.; WANG, Y.; CAO, Y.; JIA, Y.; CHEN, Z.; SHEN, J. *et al.* Hierarchical generative modeling for controllable speech synthesis. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2018. Citado nas páginas 58, 59 e 61.

HUANG, G.; LIU, Z.; Van Der Maaten, L.; WEINBERGER, K. Q. Densely connected convolutional networks. **Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017**, v. 2017-Janua, p. 2261–2269, 2017. ISSN 0002-9645. Citado nas páginas 41, 42 e 43.

HUANG, Q.; YANG, L.; HUANG, H.; WU, T.; LIN, D. Caption-supervised face recognition: Training a state-of-the-art face model without manual annotation. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2020. p. 139–155. Citado na página 40.

HUANG, X.; ACERO, A.; HON, H.-W.; REDDY, R. **Spoken language processing: A guide to theory, algorithm, and system development**. [S.l.]: Prentice hall PTR, 2001. Citado na página 55.

JARRETT, K.; KAVUKCUOGLU, K.; RANZATO, M.; LECUN, Y. What is the best multi-stage architecture for object recognition? In: IEEE. **2009 IEEE 12th international conference on computer vision**. [S.l.], 2009. p. 2146–2153. Citado na página 35.

JEFFREYS, H.; JEFFREYS, B. Change of variable in an integral. **Methods of Mathematical Physics**, Cambridge University Press Cambridge, p. 32–33, 1988. Citado na página 53.

JIA, Y.; ZHANG, Y.; WEISS, R.; WANG, Q.; SHEN, J.; REN, F.; NGUYEN, P.; PANG, R.; MORENO, I. L.; WU, Y. *et al.* Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2018. p. 4480–4490. Citado nas páginas 25, 57 e 58.

JURAFSKY, D.; MARTIN, J. H. **Speech and language processing**. [S.l.]: Pearson London:, 2014. v. 3. Citado na página 54.

KALCHBRENNER, N.; ELSEN, E.; SIMONYAN, K.; NOURY, S.; CASAGRANDE, N.; LOCKHART, E.; STIMBERG, F.; OORD, A.; DIELEMAN, S.; KAVUKCUOGLU, K. Efficient neural audio synthesis. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2018. p. 2410–2419. Citado na página 25.

KANWAR, G.; ALBERGO, M. S.; BOYDA, D.; CRANMER, K.; HACKETT, D. C.; RACANIERE, S.; REZENDE, D. J.; SHANAHAN, P. E. Equivariant flow-based sampling for lattice gauge theory. **Physical Review Letters**, APS, v. 125, n. 12, p. 121601, 2020. Citado na página 52.

KIM, J.; KIM, S.; KONG, J.; YOON, S. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. **Advances in Neural Information Processing Systems**, v. 33, 2020. Citado nas páginas 21, 23, 24, 36, 40, 48, 52, 53 e 91.

KIM, J.; KONG, J.; SON, J. Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2021. p. 5530–5540. Citado nas páginas 23, 24, 40 e 48.

KINGMA, D. P.; DHARIWAL, P. Glow: generative flow with invertible 1× 1 convolutions. In: **Proceedings of the 32nd International Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2018. p. 10236–10245. Citado na página 53.

KINGMA, D. P.; SALIMANS, T.; JOZEFOWICZ, R.; CHEN, X.; SUTSKEVER, I.; WELLING, M. Improved variational inference with inverse autoregressive flow. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2016. p. 4743–4751. Citado na página 24.

KINGMA, D. P.; WELLING, M. Auto-encoding variational bayes. **arXiv preprint arXiv:1312.6114**, 2013. Citado nas páginas 24, 49 e 52.

KINGMA, D. P.; WELLING, M. *et al.* An introduction to variational autoencoders. **Foundations and Trends® in Machine Learning**, Now Publishers, Inc., v. 12, n. 4, p. 307–392, 2019. Citado nas páginas 50 e 51.

KLAMBAUER, G.; UNTERTHINER, T.; MAYR, A.; HOCHREITER, S. Self-normalizing neural networks. In: **Proceedings of the 31st International Conference on Neural Information Processing Systems**. [S.l.: s.n.], 2017. p. 972–981. Citado na página 35.

KOBYZEV, I.; PRINCE, S.; BRUBAKER, M. Normalizing flows: An introduction and review of current methods. **IEEE Computer Architecture Letters**, IEEE Computer Society, n. 01, p. 1–1, 2020. Citado nas páginas 52 e 53.

KONG, J.; KIM, J.; BAE, J. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. **Advances in Neural Information Processing Systems**, v. 33, 2020. Citado na página 136.

KRIMAN, S.; BELIAEV, S.; GINSBURG, B.; HUANG, J.; KUCHAIEV, O.; LAVRUKHIN, V.; LEARY, R.; LI, J.; ZHANG, Y. Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions. In: IEEE. **ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2020. p. 6124–6128. Citado nas páginas 36 e 40.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2012. p. 1097–1105. Citado na página 40.

____. Imagenet classification with deep convolutional neural networks. **Communications of the ACM**, ACM New York, NY, USA, v. 60, n. 6, p. 84–90, 2017. Citado na página 36.

KUMAR, M.; BABAEIZADEH, M.; ERHAN, D.; FINN, C.; LEVINE, S.; DINH, L.; KINGMA, D. Videoflow: A flow-based generative model for video. **arXiv preprint arXiv:1903.01434**, v. 2, n. 5, 2019. Citado na página 52.

KUMAR, N.; GOEL, S.; NARANG, A.; LALL, B. Normalization Driven Zero-Shot Multi-Speaker Speech Synthesis. In: **Proc. Interspeech 2021**. [S.l.: s.n.], 2021. p. 1354–1358. Citado na página 25.

KYLE, K. K. J. F. S.; JOSE, K. A. C. Y. B.; SOTELO, S. M. Char2wav: End-to-end speech synthesis. In: **International Conference on Learning Representations, workshop**. [S.l.: s.n.], 2017. Citado nas páginas 21 e 23.

LAPTEV, A.; KOROSTIK, R.; SVISCHEV, A.; ANDRUSENKO, A.; MEDENNIKOV, I.; RYBIN, S. You do not need more data: improving end-to-end speech recognition by text-to-speech data augmentation. In: IEEE. **2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)**. [S.l.], 2020. p. 439–444. Citado nas páginas 25, 26, 59, 109 e 137.

LAX, P. D. Change of variables in multiple integrals. **The American mathematical monthly**, Taylor & Francis, v. 106, n. 6, p. 497–501, 1999. Citado na página 53.

LECUN, Y.; KANTER, I.; SOLLA, S. A. Second order properties of error surfaces: Learning time and generalization. In: **Advances in neural information processing systems**. [S.l.: s.n.], 1991. p. 918–924. Citado na página 34.

LECUN, Y. *et al.* Generalization and network design strategies. **Connectionism in perspective**, Elsevier Zurich, Switzerland, v. 19, p. 143–155, 1989. Citado na página 36.

LI, J.; GADDE, R.; GINSBURG, B.; LAVRUKHIN, V. Training neural speech recognition systems with synthetic speech augmentation. **arXiv preprint arXiv:1811.00707**, 2018. Citado nas páginas 25, 26 e 58.

LI, J.; LAVRUKHIN, V.; GINSBURG, B.; LEARY, R.; KUCHAIEV, O.; COHEN, J. M.; NGUYEN, H.; GADDE, R. T. Jasper: An end-to-end convolutional neural acoustic model. **Proc. Interspeech 2019**, p. 71–75, 2019. Citado nas páginas 21, 22, 36 e 54.

LI, N.; LIU, S.; LIU, Y.; ZHAO, S.; LIU, M. Neural speech synthesis with transformer network. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2019. v. 33, p. 6706–6713. Citado nas páginas 23, 24 e 48.

LI, S.; OUYANG, B.; LI, L.; HONG, Q. Light-tts: Lightweight multi-speaker multi-lingual text-to-speech. In: IEEE. **ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2021. p. 8383–8387. Citado nas páginas 26 e 62.

LIN, Z.; FENG, M.; SANTOS, C. N. d.; YU, M.; XIANG, B.; ZHOU, B.; BENGIO, Y. A structured self-attentive sentence embedding. **arXiv preprint arXiv:1703.03130**, 2017. Citado na página 47.

LIU, L.; HAMILTON, W. L.; LONG, G.; JIANG, J.; LAROCHELLE, H. A universal representation transformer layer for few-shot image classification. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2020. Citado na página 48.

LOIZOU, P. C. **Speech enhancement: theory and practice**. [S.l.]: CRC press, 2013. Citado nas páginas 54 e 55.

LUONG, M.-T.; PHAM, H.; MANNING, C. D. Effective approaches to attention-based neural machine translation. In: **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**. [S.l.: s.n.], 2015. p. 1412–1421. Citado na página 46.

MAAS, A. L.; HANNUN, A. Y.; NG, A. Y. Rectifier nonlinearities improve neural network acoustic models. In: **Proc. ICML**. [S.l.: s.n.], 2013. v. 30, n. 1. Citado na página 35.

MAJUMDAR, S.; BALAM, J.; HRINCHUK, O.; LAVRUKHIN, V.; NOROOZI, V.; GINSBURG, B. **Citrinet: Closing the Gap between Non-Autoregressive and Autoregressive End-to-End Models for Automatic Speech Recognition**. 2021. Citado nas páginas 21 e 23.

MAKHZANI, A.; FREY, B. K-sparse autoencoders. **arXiv preprint arXiv:1312.5663**, 2013. Citado na página 51.

MALIK, M.; MALIK, M. K.; MEHMOOD, K.; MAKHDOOM, I. Automatic speech recognition: a survey. **Multimedia Tools and Applications**, Springer, v. 80, n. 6, p. 9411–9457, 2021. Citado na página 22.

MALLAT, S. Understanding deep convolutional networks. **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, The Royal Society Publishing, v. 374, n. 2065, p. 20150203, 2016. Citado na página 41.

MARQUIAFÁVEL, V.; BOKAN, A.; ZAVAGLIA, C. Petrus: A rulebased grapheme-to-phone converter for Brazilian Portuguese. v. 11, 2014. Citado na página 55.

MAZZA, L. O. Aplicacao de redes neurais convolucionais densamente conectadas no processamento digital de imagens para remocao de ruído gaussiano. 2017. Citado na página 38.

MEDSKER, L. R.; JAIN, L. Recurrent neural networks. 2001. Citado na página 38.

MIAO, C.; SHUANG, L.; LIU, Z.; MINCHUAN, C.; MA, J.; WANG, S.; XIAO, J. Efficienttts: An efficient and high-quality text-to-speech architecture. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2021. p. 7700–7709. Citado na página 24.

MIKOLOV, T.; KARAFIÁT, M.; BURGET, L.; ČERNOCKỲ, J.; KHUDANPUR, S. Recurrent neural network based language model. In: **Eleventh Annual Conference of the International Speech Communication Association**. [S.l.: s.n.], 2010. Citado na página 38.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2013. p. 3111–3119. Citado na página 38.

MINSKY, M.; SELFRIDGE, O. G. **Learning in random nets**. [S.l.]: MIT Lincoln Laboratory, 1960. v. 46. Citado na página 31.

MISRA, D. Mish: A self regularized non-monotonic neural activation function. **arXiv preprint arXiv:1908.08681**, 2019. Citado na página 35.

MO, H.; CHEN, B.; LUO, W. Fake faces identification via convolutional neural network. In: **Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security**. [S.l.: s.n.], 2018. p. 43–47. Citado na página 36.

NEKVINDA, T.; DUŠEK, O. One model, many languages: Meta-learning for multilingual text-to-speech. **Proc. Interspeech 2020**, p. 2972–2976, 2020. Citado nas páginas 26, 60 e 61.

NG, A. *et al.* Sparse autoencoder. **CS294A Lecture notes**, v. 72, n. 2011, p. 1–19, 2011. Citado na página 51.

NIELSEN, M. A. **Neural networks and deep learning**. [S.l.]: Determination Press, 2015. Citado nas páginas 34, 35, 36, 37, 38 e 41.

PANAYOTOV, V.; CHEN, G.; POVEY, D.; KHUDANPUR, S. Librispeech: an asr corpus based on public domain audio books. In: IEEE. **Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on**. [S.l.], 2015. p. 5206–5210. Citado nas páginas 40 e 58.

PARIKH, A.; TÄCKSTRÖM, O.; DAS, D.; USZKOREIT, J. A decomposable attention model for natural language inference. In: **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**. [S.l.: s.n.], 2016. p. 2249–2255. Citado na página 47.

PARK, D. S.; CHAN, W.; ZHANG, Y.; CHIU, C.-C.; ZOPH, B.; CUBUK, E. D.; LE, Q. V. Specaugment: A simple data augmentation method for automatic speech recognition. **Proceedings INTERSPEECH 2019**, p. 2613–2617, 2019. Citado na página 180.

PARK, K.; MULC, T. Css10: A collection of single speaker speech datasets for 10 languages. **Proc. Interspeech 2019**, p. 1566–1570, 2019. Citado na página 62.

PASCANU, R.; GULCEHRE, C.; CHO, K.; BENGIO, Y. How to construct deep recurrent neural networks: Proceedings of the second international conference on learning representations (iclr 2014). In: **2nd International Conference on Learning Representations, ICLR 2014**. [S.l.: s.n.], 2014. Citado na página 41.

PAUL, D.; PANTAZIS, Y.; STYLIANOU, Y. Speaker conditional wavernn: Towards universal neural vocoder for unseen speaker and recording conditions. **Proc. Interspeech 2020**, p. 235–239, 2020. Citado na página 25.

PAUL, D. B.; BAKER, J. M. The design for the wall street journal-based csr corpus. In: ASSOCIATION FOR COMPUTATIONAL LINGUISTICS. **Proceedings of the workshop on Speech and Natural Language**. [S.l.], 1992. p. 357–362. Citado na página 22.

PENHA, D. de P.; CASTRO, A. R. G. Convolutional neural network applied to the identification of residential equipment in non-intrusive load monitoring systems. In: **3rd International Conference on Artificial Intelligence and Applications**. [S.l.: s.n.], 2017. p. 11–21. Citado na página 38.

PETKAR, H. A review of challenges in automatic speech recognition. **International Journal of Computer Applications**, Foundation of Computer Science, v. 151, n. 3, p. 23–26, 2016. Citado na página 26.

PING, W.; PENG, K.; GIBIANSKY, A.; ARIK, S. O.; KANNAN, A.; NARANG, S.; RAIMAN, J.; MILLER, J. Deep voice 3: 2000-speaker neural text-to-speech. **Proc. ICLR**, p. 214–217, 2018. Citado nas páginas 21, 23, 24 e 36.

PLATANIOS, E. A.; SACHAN, M.; NEUBIG, G.; MITCHELL, T. M. Contextual parameter generation for universal neural machine translation. In: **EMNLP**. [S.l.: s.n.], 2018. Citado na página 61.

PONTI, M. A.; RIBEIRO, L. S. F.; NAZARE, T. S.; BUI, T.; COLLOMOSSE, J. Everything you wanted to know about deep learning for computer vision but were afraid to ask. In: IEEE. **2017 30th SIBGRAPI conference on graphics, patterns and images tutorials (SIBGRAPI-T)**. [S.l.], 2017. p. 17–41. Citado na página 21.

PONTI, M. A.; SANTOS, F. P. dos; RIBEIRO, L. S.; CAVALLARI, G. B. Training deep networks from zero to hero: avoiding pitfalls and going beyond. In: IEEE. **2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)**. [S.l.], 2021. p. 9–16. Citado nas páginas 35 e 41.

PRENGER, R.; VALLE, R.; CATANZARO, B. Waveglow: A flow-based generative network for speech synthesis. In: IEEE. **ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2019. p. 3617–3621. Citado nas páginas 52 e 53.

PURINGTON, A.; TAFT, J. G.; SANNON, S.; BAZAROVA, N. N.; TAYLOR, S. H. "Alexa is my new BFF"social roles, user satisfaction, and personification of the amazon echo. In: **Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems**. [S.l.: s.n.], 2017. p. 2853–2859. Citado na página 22.

QUINTANILHA, I. M. **End-to-End Speech Recognition Applied to Brazilian Portuguese Using Deep Learning**. Tese (Doutorado) — MSc dissertation, PEE/COPPE, Federal University of Rio de Janeiro, Rio de Janeiro, Brazil, 2017. Citado nas páginas 54 e 55.

QUINTANILHA, I. M.; BISCAINHO, L. W. P.; NETTO, S. L. Towards an end-to-end speech recognizer for portuguese using deep neural networks. **XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais**, p. 709–714, 2017. Citado na página 21.

QUINTANILHA, I. M.; NETTO, S. L.; BISCAINHO, L. W. P. An open-source end-to-end asr system for brazilian portuguese using dnns built from newly assembled corpora. **Journal of Communication and Information Systems**, v. 35, n. 1, p. 230–242, 2020. Citado nas páginas 26 e 119.

RAMACHANDRAN, P.; ZOPH, B.; LE, Q. V. Searching for activation functions. **arXiv preprint arXiv:1710.05941**, 2017. Citado na página 35.

RAMAMOORTHY, C.; SHEKHAR, S. Stochastic backpropagation: a learning algorithm for generalization problems. In: IEEE. **[1989] Proceedings of the Thirteenth Annual International Computer Software & Applications Conference**. [S.l.], 1989. p. 664–671. Citado na página 31.

RAMANI, A.; RAO, A.; VIDYA, V.; PRASAD, V. B. Automatic subtitle generation for videos. In: IEEE. **2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)**. [S.l.], 2020. p. 132–135. Citado na página 21.

RAUBER, T. W. Redes neurais artificiais. **Universidade Federal do Espírito Santo**, 2005. Citado na página 33.

RAZAVIAN, A. S.; AZIZPOUR, H.; SULLIVAN, J.; CARLSSON, S. Cnn features off-the-shelf: an astounding baseline for recognition. In: IEEE. **Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on**. [S.l.], 2014. p. 512–519. Citado na página 36.

REDFORD, M. A. **The handbook of speech production**. [S.l.]: John Wiley & Sons, 2015. Citado na página 56.

REN, J.; HU, Y.; TAI, Y.-W.; WANG, C.; XU, L.; SUN, W.; YAN, Q. Look, listen and learn—a multimodal lstm for speaker identification. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2016. v. 30, n. 1. Citado na página 38.

REN, Y.; HU, C.; TAN, X.; QIN, T.; ZHAO, S.; ZHAO, Z.; LIU, T.-Y. Fastspeech 2: Fast and high-quality end-to-end text to speech. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2020. Citado na página 24.

RESENDE, D. C. O. de; PONTI, M. A. Robust image features for classification and zero-shot tasks by merging visual and semantic attributes. **Neural Computing and Applications**, Springer, p. 1–13, 2022. Citado na página 25.

REZENDE, D.; MOHAMED, S. Variational inference with normalizing flows. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2015. p. 1530–1538. Citado na página 52.

RIBANI, M.; BOTTOLI, C. B. G.; COLLINS, C. H.; JARDIM, I. C. S. F.; MELO, L. F. C. Validation for chromatographic and electrophoretic methods. **Quimica Nova**, SciELO Brasil, v. 27, n. 5, p. 771–780, 2004. Citado na página 54.

RIBEIRO, F.; FLORÊNCIO, D.; ZHANG, C.; SELTZER, M. Crowdmos: An approach for crowdsourcing mean opinion score studies. In: IEEE. **Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on**. [S.l.], 2011. p. 2416–2419. Citado nas páginas 56 e 57.

RIBEIRO, L. S. F.; BUI, T.; COLLOMOSSE, J.; PONTI, M. Sketchformer: Transformer-based representation for sketched structure. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 14153–14162. Citado na página 48.

RIFAI, S.; VINCENT, P.; MULLER, X.; GLOROT, X.; BENGIO, Y. Contracting auto-encoders: Explicit invariance during feature extraction. In: CITESEER. **In Proceedings of the Twenty-eight International Conference on Machine Learning (ICML'11)**. [S.l.], 2011. Citado na página 51.

ROSENBERG, A.; ZHANG, Y.; RAMABHADRAN, B.; JIA, Y.; MORENO, P.; WU, Y.; WU, Z. Speech recognition with augmented synthesized speech. In: IEEE. **2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)**. [S.l.], 2019. p. 996–1002. Citado nas páginas 25, 26 e 58.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citado na página 31.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning internal representations by error propagation**. [S.l.], 1985. Citado na página 38.

RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATHY, A.; KHOSLA, A.; BERNSTEIN, M. *et al.* Imagenet large scale visual recognition challenge. **International Journal of Computer Vision**, Springer, v. 115, n. 3, p. 211–252, 2015. Citado na página 40.

RUSSELL, S. J.; NORVIG, P. **Artificial intelligence: a modern approach**. [S.l.]: Malaysia; Pearson Education Limited,, 2016. Citado nas páginas 31, 32 e 33.

SAHIDULLAH, M.; SAHA, G. Design, analysis and experimental evaluation of block based transformation in mfcc computation for speaker recognition. **Speech communication**, Elsevier, v. 54, n. 4, p. 543–565, 2012. Citado na página 55.

SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v. 26, n. 1, p. 43–49, 1978. Citado na página 22.

SCHNEIDER, S.; BAEVSKI, A.; COLLOBERT, R.; AULI, M. wav2vec: Unsupervised pretraining for speech recognition. **Proc. Interspeech 2019**, p. 3465–3469, 2019. Citado nas páginas 21, 22 e 55.

SHANG, W.; SOHN, K.; ALMEIDA, D.; LEE, H. Understanding and improving convolutional neural networks via concatenated rectified linear units. In: **International Conference on Machine Learning**. [S.l.: s.n.], 2016. p. 2217–2225. Citado na página 35.

SHEN, J.; PANG, R.; WEISS, R. J.; SCHUSTER, M.; JAITLY, N.; YANG, Z.; CHEN, Z.; ZHANG, Y.; WANG, Y.; SKERRV-RYAN, R. *et al.* Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In: IEEE. **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2018. p. 4779–4783. Citado nas páginas 21, 23, 24, 25, 38, 44 e 58.

SHI, W.; CABALLERO, J.; THEIS, L.; HUSZAR, F.; AITKEN, A.; LEDIG, C.; WANG, Z. Is the deconvolution layer the same as a convolutional layer? **arXiv preprint arXiv:1609.07009**, 2016. Citado na página 49.

SHI, Y.; YU, X.; SOHN, K.; CHANDRAKER, M.; JAIN, A. K. Towards universal representation learning for deep face recognition. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 6817–6826. Citado na página 40.

SNYDER, D.; GARCIA-ROMERO, D.; SELL, G.; POVEY, D.; KHUDANPUR, S. X-vectors: Robust dnn embeddings for speaker recognition. In: IEEE. **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2018. p. 5329–5333. Citado nas páginas 62 e 180.

SRINIVAS, A.; LIN, T.-Y.; PARMAR, N.; SHLENS, J.; ABBEEL, P.; VASWANI, A. Bottleneck transformers for visual recognition. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2021. p. 16519–16529. Citado na página 39.

SRIVASTAVA, R. K.; GREFF, K.; SCHMIDHUBER, J. Training very deep networks. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2015. p. 2377–2385. Citado na página 41.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. **Advances in Neural Information Processing Systems**, v. 27, p. 3104–3112, 2014. Citado na página 44.

SZTAHÓ, D.; SZASZÁK, G.; BEKE, A. Deep learning methods in speaker recognition: A review. **Periodica Polytechnica. Electrical Engineering and Computer Science**, Periodica Polytechnica, Budapest University of Technology and Economics, v. 65, n. 4, p. 310, 2021. Citado na página 56.

TACHIBANA, H.; UENOYAMA, K.; AIHARA, S. Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. In: IEEE. **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2018. p. 4784–4788. Citado nas páginas 21, 23, 24, 36 e 61.

TAN, H. H.; LIM, K. H. Vanishing gradient mitigation with deep learning neural network optimization. In: IEEE. **2019 7th International Conference on Smart Computing & Communications (ICSCC)**. [S.l.], 2019. p. 1–4. Citado na página 34.

TAN, X.; QIN, T.; SOONG, F.; LIU, T.-Y. A survey on neural speech synthesis. **arXiv preprint arXiv:2106.15561**, 2021. Citado nas páginas 23, 99 e 137.

TOUVRON, H.; CORD, M.; DOUZE, M.; MASSA, F.; SABLAYROLLES, A.; JÉGOU, H. Training data-efficient image transformers & distillation through attention. **arXiv preprint arXiv:2012.12877**, 2020. Citado na página 39.

TREVISO, M.; SHULBY, C.; ALUÍSIO, S. Evaluating word embeddings for sentence boundary detection in speech transcripts. In: **Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology**. Uberlândia, Brazil: Sociedade Brasileira de Computação, 2017. p. 151–160. Disponível em: <https://www.aclweb.org/anthology/W17-6618>. Citado na página 159.

TULSHAN, A. S.; DHAGE, S. N. Survey on virtual assistant: Google assistant, siri, cortana, alexa. In: SPRINGER. **International symposium on signal processing and intelligent recognition systems**. [S.l.], 2018. p. 190–201. Citado na página 21.

VALLE, R.; SHIH, K.; PRENGER, R.; CATANZARO, B. Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis. **arXiv preprint arXiv:2005.05957**, 2020. Citado nas páginas 21, 23, 24, 40, 52, 53, 54 e 91.

VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres. In: **Proceedings of the XXIX Conference on Graphics, Patterns and Images**. [S.l.: s.n.], 2016. p. 1–4. Citado na página 37.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. In: **Advances in neural information processing systems**. [S.l.: s.n.], 2017. p. 5998–6008. Citado nas páginas 24, 46, 47, 48 e 49.

VEAUX, C.; YAMAGISHI, J.; MACDONALD, K. *et al.* Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. **University of Edinburgh. The Centre for Speech Technology Research (CSTR)**, 2016. Citado na página 26.

VELICHKO, V.; ZAGORUYKO, N. Automatic recognition of 200 words. **International Journal of Man-Machine Studies**, Elsevier BV, v. 2, n. 3, p. 223–234, jul 1970. Disponível em: <https://doi.org/10.1016%2Fs0020-7373%2870%2980008-6>. Citado na página 22.

VINCENT, P.; LAROCHELLE, H.; BENGIO, Y.; MANZAGOL, P.-A. Extracting and composing robust features with denoising autoencoders. In: **Proceedings of the 25th international conference on Machine learning**. [S.l.: s.n.], 2008. p. 1096–1103. Citado na página 51.

WAN, L.; WANG, Q.; PAPIR, A.; MORENO, I. L. Generalized end-to-end loss for speaker verification. In: IEEE. **2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2018. p. 4879–4883. Citado na página 25.

WANG, Y.; SKERRY-RYAN, R.; STANTON, D.; WU, Y.; WEISS, R. J.; JAITLY, N.; YANG, Z.; XIAO, Y.; CHEN, Z.; BENGIO, S. *et al.* Tacotron: A fully end-to-end text-to-speech synthesis model. **arXiv preprint arXiv:1703.10135**, 2017. Citado nas páginas 21, 23, 24 e 38.

WANG, Y.; STANTON, D.; ZHANG, Y.; RYAN, R.-S.; BATTENBERG, E.; SHOR, J.; XIAO, Y.; JIA, Y.; REN, F.; SAUROUS, R. A. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2018. p. 5180–5189. Citado na página 58.

WANGKEEREE, N.; BOONKRONG, S. Finding a suitable threshold value for an iris-based authentication system. **International Journal of Electrical and Computer Engineering**, IAES Institute of Advanced Engineering and Science, v. 9, n. 5, p. 3558, 2019. Citado na página 56.

WATANABE, S.; HORI, T.; KARITA, S.; HAYASHI, T.; NISHITOBA, J.; UNNO, Y.; SOPLIN, N.-E. Y.; HEYMANN, J.; WIESNER, M.; CHEN, N. *et al.* Espnet: End-to-end speech processing toolkit. **Proc. Interspeech 2018**, p. 2207–2211, 2018. Citado na página 59.

WENG, L. Attention? attention! **lilianweng.github.io/lil-log**, 2018. Disponível em: <http://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html>. Citado na página 44.

____. Flow-based deep generative models. **lilianweng.github.io/lil-log**, 2018. Disponível em: <http://lilianweng.github.io/lil-log/2018/10/13/flow-based-deep-generative-models.html>. Citado nas páginas 53 e 54.

____. From autoencoder to beta-vae. **lilianweng.github.io/lil-log**, 2018. Disponível em: <http://lilianweng.github.io/lil-log/2018/08/12/from-autoencoder-to-beta-vae.html>. Citado nas páginas 50 e 51.

WU, F.; FAN, A.; BAEVSKI, A.; DAUPHIN, Y.; AULI, M. Pay less attention with lightweight and dynamic convolutions. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2018. Citado na página 62.

YANG, X.; JIA, X.; GONG, D.; YAN, D.-M.; LI, Z.; LIU, W. Larnet: Lie algebra residual network for face recognition. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2021. p. 11738–11750. Citado na página 40.

YU, D.; DENG, L. **AUTOMATIC SPEECH RECOGNITION.** [S.l.]: Springer, 2016. Citado nas páginas 21, 39 e 40.

ZE, H.; SENIOR, A.; SCHUSTER, M. Statistical parametric speech synthesis using deep neural networks. In: IEEE. **2013 ieee international conference on acoustics, speech and signal processing**. [S.l.], 2013. p. 7962–7966. Citado na página 23.

ZEN, H.; DANG, V.; CLARK, R.; ZHANG, Y.; WEISS, R. J.; JIA, Y.; CHEN, Z.; WU, Y. Libritts: A corpus derived from librispeech for text-to-speech. **Proc. Interspeech 2019**, p. 1526–1530, 2019. Citado nas páginas 26 e 40.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. Dive into deep learning. 2020. **URL https://d2l. ai**, 2020. Citado nas páginas 39, 44, 45, 46 e 47.

ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. mixup: Beyond empirical risk minimization. **International Conference on Learning Representations**, 2018. Disponível em: <https://openreview.net/forum?id=r1Ddp1-Rb>. Citado na página 180.

ZHANG, Y.-J.; PAN, S.; HE, L.; LING, Z.-H. Learning latent representations for style control and transfer in end-to-end speech synthesis. In: IEEE. **ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)**. [S.l.], 2019. p. 6945–6949. Citado nas páginas 26, 51 e 61.

ZHOU, T.; ZHAO, Y.; WU, J. Resnext and res2net structures for speaker verification. In: IEEE. **2021 IEEE Spoken Language Technology Workshop (SLT)**. [S.l.], 2021. p. 301–307. Citado na página 40.

APÊNDICE A

APLICAÇÕES EM PROCESSAMENTO DE FALA

A.1 Evaluating Sentence Segmentation in Different Datasets of Neuropsychological Language Tests in Brazilian Portuguese

Título:	**Evaluating Sentence Segmentation in Different Datasets of Neuropsychological Language Tests in Brazilian Portuguese**
Autores:	**Edresson Casanova, Marcos V. Treviso, Lilian C. Hübner e Sandra M. Aluísio**
Ano:	**2020**
Conferência:	**The 12th Language Resources and Evaluation Conference (LREC 2020)**
Situação:	**Publicado**

Contribuições relevantes do artigo:

- Propõe melhorias na arquitetura baseada em Recursive Convolutional Neural Networks (RCNN) proposta por Treviso, Shulby e Aluísio (2017) para detecção de limite de sentença em cenários de fala comprometida, apresentando três novas arquiteturas;

- Apresenta uma análise em busca de um detector de limite de sentença genérico, que obtenha um bom desempenho em narrativas com estímulos visuais bem como em narrativas com estímulos orais;

- Apresenta e disponibiliza publicamente três datasets para detecção de limite de sentença em cenários de fala comprometida.

Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 2605–2614
Marseille, 11–16 May 2020

Evaluating Sentence Segmentation in Different Datasets of Neuropsychological Language Tests in Brazilian Portuguese

Edresson Casanova[1], Marcos V. Treviso[2*], Lilian C. Hübner[3], Sandra M. Aluísio[1]
[1]University of São Paulo, [2]Instituto de Telecomunicações, [3]Pontifical Catholic University of Rio Grande do Sul
edresson@usp.br, marcos.treviso@lx.it.pt, lilian.hubner@pucrs.br, sandra@icmc.usp.br

Abstract

Automatic analysis of connected speech by natural language processing techniques is a promising direction for diagnosing cognitive impairments. However, some difficulties still remain: the time required for manual narrative transcription and the decision on how transcripts should be divided into sentences for successful application of parsers used in metrics, such as Idea Density, to analyze the transcripts. The main goal of this paper was to develop a generic segmentation system for narratives of neuropsychological language tests. We explored the performance of our previous single-dataset-trained sentence segmentation architecture in a richer scenario involving three new datasets used to diagnose cognitive impairments, comprising different stories and two types of stimulus presentation for eliciting narratives — visual and oral — via illustrated story-book and sequence of scenes, and by retelling. Also, we proposed and evaluated three modifications to our previous RCNN architecture: (i) the inclusion of a Linear Chain CRF; (ii) the inclusion of a self-attention mechanism; and (iii) the replacement of the LSTM recurrent layer by a Quasi-Recurrent Neural Network layer. Our study allowed us to develop two new models for segmenting impaired speech transcriptions, along with an ideal combination of datasets and specific groups of narratives to be used as the training set.

Keywords: Sentence Segmentation, Impaired Speech, Neuropsychological Language Tests

1. Introduction

Language assessment has been shown to be an efficient complementary tool for detecting cognitive and neuropsychological disorders, therefore present in most tests, tasks and batteries that evaluate cognitive processes. For example, neuropsychological language tests are an important tool for diagnosing individuals with significant depression in Alzheimer's disease (AD) (Fraser et al., 2016), to differentiate between Mild Cognitive Impairment (MCI) and AD (Drummond et al., 2015), to differentiate between AD and other neurodegenerative dementias (Yancheva et al., 2015; Beltrami et al., 2018) and to differentiate variants of neurodegenerative dementias, such as in Primary Progressive Aphasia (PPA) (Fraser et al., 2014).

Language assessment has been performed mainly by using discursive production in which narratives are largely used, since they are a natural form of communication and favor the observation of the patient's functionality in everyday life (Tillas, 2015). The discourse tasks used to assess the narrative productions of elder individuals are often based on: (i) an illustrated story book without a text (e.g. Cinderella), (ii) an immediate and delayed retelling of a story orally presented, or (iii) a single scene or a sequence of scenes, presented on pictures, of a common event that occurs in daily life.

With regard to specific batteries used to evaluate language in discourse tasks, we can cite a few (Wechsler, 1997; Bayles and Tomoeda, 1993; Goodglass et al., 1983; Hübner et al., 2019). Discourse tasks that require some degree of memorization are usually included in verbal memory tests. This is the case of the Logical Memory Subtest task from the Wechsler Memory Scale, used for assessing episodic memory (Wechsler, 1997). In this task, an individual reproduces a story immediately after listening to it (immediate recall); thirty minutes later, subjects are asked to recall the story again (delayed recall). The retellings are transcribed for further analysis. The higher the number of recalled elements of the narrative, the higher the memory score. This procedure is also used in the Arizona Battery for Communication Disorders of Dementia (ABCD) (Bayles and Tomoeda, 1993). The single-scene description task called "The Cookie Theft Picture" is part of the Boston Diagnostic Aphasia Examination (BDAE) (Goodglass et al., 1983). The Cookie-Theft picture has been shown to be clinically relevant in identifying linguistic deficits in Alzheimer's disease patients given the importance of using visual stimuli when evaluating individuals of this group. The Cinderella story is also very widely used in the assessment of aphasia and some types of dementia. (Fraser et al., 2015) and (Aluisio et al., 2016) based their work on the story of Cinderella. Participants were given a sequenced picture book (without words) to remind them of the story; then they were asked to tell the story in their own words. The narrative samples were then transcribed by trained annotators.

A challenge in choosing the type of neuropsychological assessment of individuals with AD and MCI is the use of a battery that can distinguish these individuals. ABCD appears as an option since it is capable of detecting mild stage AD, while the *Bateria de Avaliação da Linguagem no Envelhecimento* (BALE) (Battery of Language Assessment in Aging, in English) (Hübner et al., 2019) was developed for application to patients with different educational levels, including illiterate ones, a very common condition among Brazilian elders.

All the studies cited above confirm that the automatic analysis of connected speech by natural language processing techniques (NLP) is a promising direction for diagnosing cognitive impairments. However, some difficulties still re-

[*] Work carried out during the master's course at the University of São Paulo.

main: the time required for manual narrative transcription and the decision on how transcripts should be divided into sentences for (i) extracting narrative recall scores automatically from semantic similarity methods applied to sentences — the shorter the sentences, the better the method response (see (Borges dos Santos and Aluísio, 2020)) and (ii) the successful application of parsers used in metrics to analyze the transcripts. One of these metrics is called Idea Density and was originally proposed as a way of measuring the memory load of narratives, by representing the underlying content of the text as a series of semantic units, called propositions or ideas. The method proposed by da Cunha et al. (2015) is a rule-based system acting upon dependency trees, strongly depending on a robust parser.

Growing consensus in the NLP area indicates that in order to have fully automated systems for diagnosing cognitive impairments, an NLP pipeline must use an Automatic Speech Recognition (ASR) system. Although to build a high performance ASR for pathological language can be a long term research, we will ignore the issue (i) for now. If we have a manual transcription or an automatically generated one, we still have to detect sentence boundaries, therefore, we will focus on this task in this paper.

Since the majority of studies on diagnosing cognitive impairments by NLP methods deal with English-speaking patients (Filiou et al., 2019), in this study we will evaluate the Brazilian Portuguese (BP) language in order to contribute with datasets and studies to develop automatic analysis of connected speech in BP. Our motivation for this study was to explore the performance of a single-dataset-trained sentence segmentation architecture in a richer scenario involving three new datasets. Therefore, here, we evaluate four datasets used to diagnose cognitive impairments (see Section 3), comprising different stories and two type of stimulus presentation for eliciting narratives: (i) oral stimuli presentation with retelling, where sequencing discourse marks, such as "e","aí", "daí" and "então" (and, then, in English) and confirmatory discourse marks "né" and "ok" (ok, in English) are frequent and (ii) visual stimuli, via both illustrated story book and sequence of scenes of a common event, where deictic expressions (place deixis) are pervasive, such as "aqui" and "aí" (here) and "ali" and "lá" (there), besides presenting sequencing discourse marks and confirmatory discourse.

Figure 1 shows the result of a manual transcription in BP of a narrative of the ABCD story telling task, which presents a story about a woman who is unaware of having lost her wallet while doing the shopping; she then receives a call from a little girl who found the wallet. As we can see in (a) the transcript without punctuation prevents the direct application of NLP methods that rely on sentence segmentation for the correct use of tools as taggers and parsers. These tools are used to implement metrics of syntactic complexity, basic counts of PoS tags and to analyze other levels of language to diagnose cognitive impairments.

(a) **ahm** uma senhora **foi fazer compras no me** foi no mercado não lembrava o local **no me** fazer compras e quando ela foi pagar a conta no caixa percebeu que estava sem a carteira aí ela **foi deixou a mercadoria** não levou a mercadoria voltou para casa chegando em casa toca o telefone era uma garotinha avisando ela que que tinha achado a carteira **é isso tem mais coisa não cortei eu resumi o que eu ouvi**

(b) ahm uma senhora foi fazer compras no me foi no mercado. não lembrava o local . no me fazer compras . e quando ela foi pagar a conta no caixa percebeu que estava sem a carteira . aí ela foi deixou a mercadoria . não levou a mercadoria . voltou para casa . chegando em casa toca o telefone . era uma garotinha avisando ela que que tinha achado a carteira . é isso . tem mais coisa . não cortei . eu resumi o que eu ouvi .

Figure 1: (a) Narrative transcribed where there is no punctuation or capitalization, besides presenting several disfluencies, such as unlexicalized filled pauses, restarts and patient's comments, shown in bold. (b) Narrative manually segmented of a retelling task using The Wallet Story.

When the architecture developed in the project DeepBonDD[1] was trained with The Cinderella Story dataset (a production task elicited via an illustrated story book) and evaluated with the other three datasets analyzed in this paper, F_1 values were much lower than the original work on Cinderella (Treviso et al., 2017a) (Table 1). The average of F_1 in the three datasets, for all classes, is 0.59. In the original evaluation with training and cross-validation testing in the same dataset, the best F_1 value for Controls was 0.76, for MCIs, 0.74, and for ADs, 0.66. However, Table 1 shows that, in general, for sentence segmentation, more data is beneficial, independently of task and topic of datasets.

Given this motivation scenario, where the main goal was to develop a robust and generic segmentation system for narratives of neuropsychological language tests, the present study tries to answer three questions:

1. Would modifications to the Recursive Convolutional Neural Networks (RCNN) architecture proposed by (Treviso et al., 2017a) have better performance on sentence boundary detection of new tasks and story topics? (Section 2 presents details about the RCNN architecture proposed by (Treviso et al., 2017a) and Section 4 presents the three modifications evaluated here);

2. Are there particularities in tasks that elicit narratives with visual stimuli not present on those elicited with oral stimuli (and vice versa), requiring specific sentence segmentation detectors for each task? (Section 5 presents our experiments on this issue.)

3. Do the story topics of language tests and the group of elders of a dataset negatively impact sentence segmentation detectors, also requiring specific sentence segmentation detectors? (Section 6 presents our evaluations on this issue.)

Section 2 presents a literature review on sentence segmentation, focusing on spontaneous and impaired speech. Section 7 presents the discussions and the contributions towards building a full-fledged system for automating neuropsychological tests in Portuguese.

2. Related Work on Sentence Segmentation for Impaired Speech

Due to the increasing usage of ASR systems, which usually output a stream of tokens without any capitalization

[1] https://github.com/mtreviso/deepbond

Training set	Test set	MCIs	Controls	ADs	Average
Cinderella - Same class	The Dog Story	0.43	0.54	0.56	0.51
Cinderella - All classes	The Dog Story	**0.58**	0.59	0.54	0.57
Cinderella - Same class	Lucia	n/a	0.67	0.54	0.60
Cinderella - All classes	Lucia	n/a	**0.66**	**0.62**	0.64
Cinderella - Same class	Wallet	0.54	0.56	n/a	0.55
Cinderella - All classes	Wallet	0.57	0.53	n/a	0.55

Table 1: Robustness tests in terms of F_1 using the original RCNN model trained on the Cinderella dataset. "Same class" means that the method was trained only with the same specific class used for testing. "All classes" means that data from all classes were used for training. "n/a" entries denote that the tested dataset does not have examples for that specific class.

or punctuation symbols, methods for detecting sentences boundaries were applied to solve the need of subsequent tools (such as taggers and parsers) in an NLP pipeline.

Previous methods, such as Decision Trees combined with Language Models (Shriberg et al., 2000; Liu et al., 2005b; Christensen et al., 2006), Maximum entropy models (Batista et al., 2012) and CRFs (Khomitsevich et al., 2015; Fraser et al., 2015) were applied both for prepared and spontaneous speech. These methods rely on lexical and prosodic clues (e.g. pitch, energy and pauses) in order to detect the correct position of a sentence boundary. For instance, the CRF method proposed by (Fraser et al., 2015) uses lexical, prosodic and Part-of-Speech (PoS) tags as features to segment speech from elder people with aphasia. They found that by using all these features together the model yields better results and the mistakes made by the model don't affect much the syntactic structure of the segmented transcript.

More recently, Recurrent and Convolutional Neural Networks were employed for both types of speech and achieved good results by using word embeddings as the lexical representation of words (Tilk and Alumäe, 2016; Che et al., 2016; Treviso et al., 2017a; González-Gallardo and Torres-Moreno, 2018), suggesting that deep neural networks can be successfully applied for this task.

Prosodic features have been shown to be very effective to discriminate between different types of sentence boundaries and in general their usage reflects better results (Shriberg et al., 2009; Huang et al., 2014; Khomitsevich et al., 2015). However, to put prosodic features into practice we need alignments between the audio and its transcription, which is hard to obtain mainly due to the low quality of the recordings. This problem is even more critical for impaired speech, where patients with cognitive impairment usually produce a narrative in which sentences are not syntactically well-formed, words are pronounced in a way that modifies their original morphology, and utterances have low prosody quality (elder speakers with a very low voice volume). Even long pauses are not always an indication of sentence boundaries due to word-finding difficulty of elders (Fraser et al., 2015). Therefore, prosody is hardly ever a good feature for the classifier.

Recent studies show that by using only lexical clues it is possible to achieve a comparable performance with methods that use prosodic features altogether (Klejch et al., 2016; Klejch et al., 2017). Moreover, by leveraging transfer learning techniques it is possible to reduce the drop in performance even more. For example, the method developed by (Treviso et al., 2017a), which was evaluated with different types of word embeddings, showed that by using a good word embedding representation it is possible to achieve similar results in the SOTA. Their method consists of a combination of Recurrent and Convolutional Neural Networks (RCNN). Its complete architecture is composed by the following four components:

1. An embedding layer maps words to dense vectors representations;
2. These vectors are fed to a convolutional layer that is responsible for the automatic extraction of new features depending on neighboring words;
3. The new extracted features are passed to a Bi-LSTM to capture long range dependencies; and
4. The output of the recurrent layer is projected to a binary output where a softmax operation is calculated, giving the probability of whether or not the word precedes a sentence boundary.

Since the number of sentence boundaries is much lower than the one of non-sentence boundaries, this is categorized as an unbalanced classification problem. To deal with this, the RCNN gives a higher weight to the minority class in the objective function.

The main drawback of the RCNN model is its large number of parameters combined with the small amount of training data (usual in clinical data), which usually leads to overfitting, and therefore careful regularization strategies have to be employed. In practice, we found that for narratives of new story topics the RCNN model is not able to often detect good sentence boundaries, relying on discourse marks and therefore generating very small sentences, with a main verb as the manual segmentation does.

By inspecting the errors of the RCNN model, we also found that its most common mistakes are related to deictics, sequencing and confirmatory marks preceding (e.g. "lá", "né", "ok") and succeeding sentence boundaries (e.g. "aí", "daí", "então"). Although it was shown that these errors do not affect too much the syntactic structure of the sentence, they could be easily captured by considering lexical clues more effectively (Treviso et al., 2018). Examples of places where the model should have put a sentence boundary but it missed it (i.e. false negatives) are shown in Fig. 2

(a) menino que foi na cidade . aí tá caminhando na rua . daí viu as pessoas lá . daí encontrou um cachorrinho . o cachorrinho tava perdido . chegando lá a mãe abriu a porta . e ele pediu pra mãe deixar o cachorro lá . morar com ele lá . aí arrumou até uma casinha pro cachorrinho . aí ela consentiu ele deixar até fazer uma casinha pro cachorro .

(b) menino que foi na **cidade aí** tá caminhando na **rua daí** viu as pessoas **lá daí** encontrou um **cachorrinho o** cachorrinho tava perdido . chegando lá a mãe abriu a porta . e ele pediu pra mãe deixar o cachorro **lá morar** com ele lá . aí arrumou até uma casinha pro cachorrinho . aí ela consentiu ele deixar até fazer uma casinha pro cachorro .

Figure 2: (a) Manual segmentation of a narrative elicited via a sequence of scenes of a common event (The Dog Story). (b) Example of errors made by the RCNN model; slots where a sentence boundary should have been put are shown in bold.

3. Datasets

Four datasets were used to train our models (Sections 3.1, 3.2 and 3.3). As a preprocessing step we removed capitalization information and in order to simulate high-quality ASR, we left all speech disfluencies intact. Demographic information of participants and statistics about the narratives of our study are presented in Table 2. Datasets of Neuropsychological Language Tests are typically small, as can be seen in Table 2. Table 2 shows the uniform mean length of sentences of three datasets (The Wallet Story, The Dog Story and The Cinderella Story), with regards to the groups MCI and Controls. This is an interesting feature to train/test a model, using a large dataset which combines several stories. Cinderella's mean length of narratives is very long, while both retellings produce short narratives.

In the four datasets of this study, we have segmented sentences using prosodic, syntactic and semantic knowledge, to create short sentences, with a sole idea, i.e. with an unique main verb. Therefore, coordinated sentences where divided. Although this decision has an impact on certain syntactic metrics, such as the number of sentences with coordination and the length of sentences, it makes possible for parsers to function properly over impaired speech. The manual sentence segmentation was performed by peers in two datasets, and both kappa values are very high (Landis and Koch, 1977): the kappa value for the Cinderella Story was 0.84 (almost perfect agreement), and for the Dog Story was 0.77 (substantial agreement). Therefore, the remaining annotation was performed by a sole annotator.

3.1. The Wallet Story from ABCD

ABCD is a standardized test battery for the comprehensive assessment and screening of dementia. It includes 17 subtests that evaluate linguistic expression, linguistic comprehension, verbal episodic memory via immediate/delayed recall of stories, visuospatial construction, and mental status. The subtest which is important for our study is the evaluation of the episodic memory, which is composed of the immediate and late retelling of a memorized story from (Bayles and Tomoeda, 1993), the Wallet Story. This story was translated and adapted to BP by Danielle Rüegg, Isabel Maranhão de Carvalho, Leticia Lessa Mansur and Márcia Radanovic, and was administered and collected by the team coordinated by Professor Dr. Leticia Lessa Mansur at the University of São Paulo Medical School to 23 elders with MCI and 12 healthy aging adults; totaling 70 narratives. This test has 17 units of information, with possible alternatives, with 17 being its maximum score.

3.2. The Cinderella Dataset

The Cinderella dataset consists of spontaneous speech narratives produced during a test to elicit narrative discourse with visual stimuli, using a book composed of sequenced pictures portraying the the Cinderella Story. In the test, the examinee verbally tells the story to the examiner based on the pictures. The narrative is recorded and manually transcribed by a trained annotator who scores the narrative by counting the number of recalled propositions/units of information; there are 28 informational units to be recalled, presented in 23 pictures. This dataset consists of 60 narratives from BP speakers (20 controls, 20 with AD, and 20 with amnestic MCI), diagnosed and collected at the University of São Paulo Medical School and also used in (Toledo et al., 2017; Aluisio et al., 2016; Treviso et al., 2017b; Treviso et al., 2017a).

3.3. The Dog Story and Lucia Story Datasets from BALE

BALE is a standardized battery with norms for the healthy elders Brazilian population illiterate, with low (2 to 8 years of schooling) and high (9 years or more) education, from 60 to 90 years old, described in (Hübner et al., 2019). BALE provides the academy and clinicians with standardized and validated tasks, filling an important gap in terms of tasks validated for BP, specially at the discourse level. It was conceived by the adaptation of other tasks nationally and internationally used to test language impairment mainly in AD, following psycholinguistic criteria, including imageability, frequency, animability, extension, among others, such as cultural issues. It consists of 10 linguistic tasks, assessing from the word level, in the naming task, for example, to the discourse level. One of its differentials is to evaluate discourse in four types of narrative texts, especially at the production level, but with the implicit textual comprehension as well. This battery was chosen because its aim is to allow for its administration to elder people who are illiterate and/or of low educational level, who represent the majority of the aged sample assisted by the public health system in Brazil. The Dog Story and Lucia Story are two of the four narrative texts from the BALE instrument. The Dog Story dataset is composed of transcriptions from the oral narrative production test based on the presentation of a set of seven pictures telling a story of a boy who hides a dog that he found on the street, based on the story of LeBooeuf (Le Boeuf, 1976). This dataset consists of 106 narrative texts from BP speakers, including 82 healthy aging adults, 12 with AD, and 12 with MCI. BALE also includes a task of retelling and text comprehension of an orally presented story called Lucia Story. This test has 24 units of information, with possible alternatives, with 24 being its maximum score. This retelling test was administered to 9 Alzheimer's individuals and 80 healthy aging adults. Both datasets were collected by the team coordinated by Professor Dr. Lilian Cristine Hübner of the School of Humanities at the Pontifical Catholic University of Rio Grande do Sul (PUCRS).

Stories	Groups	Nb. of Subjects	Age	Years of Education	Nb. of Sentences	Mean length of Sentences (σ)	Mean length of Narratives (σ)
The Wallet Story	MCI	23	62+	4+	376	7.45 (±3.99)	60.87 (±17.22)
	Control	12	55+	4+	184	7.70 (±4.29)	59.00 (±14.41)
The Lucia Story	Control	80	63+	2+	564	8.44 (±5.57)	59.51 (±21.36)
	AD	9	68+	1+	39	6.54 (±5.59)	28.33 (±18.37)
The Dog Story	MCI	12	57+	2+	173	8.26 (±4.43)	119.08 (±41.61)
	Control	82	60+	0+	1170	8.44 (±5.10)	120.46 (±51.65)
	AD	12	59+	0+	153	7.60 (±5.50)	96.92 (±37.56)
The Cinderella Story	MCI	20	60+	3+	618	12.38 (±7.40)	404.80 (±198.40)
	Control	20	60+	3+	654	12.79 (±7.23)	395.25 (±210.33)
	AD	20	60+	3+	794	9.83 (±7.00)	390.30 (±285.91)

Table 2: Statistics of narratives and of the Control and patient groups; the first two datasets are based on retellings and the last ones are based on sequenced figures.

4. Exp. I: Would New Architectures Have Better Performance?

Here, we propose and evaluate three modifications to the RCNN architecture developed by (Treviso et al., 2017a), namely: (i) the inclusion of a Linear Chain CRF to capture pairwise dependencies between labels; (ii) the inclusion of a self attention mechanism with the aim of capturing very long dependencies; and (iii) the replacement of the LSTM recurrent layer by a Quasi-Recurrent Neural Networks (QRNN) (Bradbury et al., 2016) layer in order to reduce the number of trainable parameters.

A CRF model can be helpful since we have sequences of labels that are very unlikely (or even impossible) to happen, such as a sequence of three sentence boundaries one after the other: `B B B`. Furthermore, by applying Viterbi decoding we can seek the best sequence of labels taking into account these transition likelihoods. In contrast to RNNs that have to remember decisions across a very long stream of tokens, attention mechanisms can access distant positions in the input at any moment, therefore they can be very helpful to learn very long dependencies. Despite having less parameters than LSTMs, a QRNN layer is based on convolutional and pooling operations, which can be computed effectively in parallel and, as a result, decrease both training and inference time.

To choose the best architectures, several configurations were evaluated using greedy search on the hyperparameters found in Table 3. We used a 5-fold cross-validation on the MCI class set of The Cinderella Story dataset. The choice of evaluating only on the MCI class was due to the high demanding time of running all experiments in all four datasets and the fact that this class represents cognitive impairment between the characteristics of Controls and ADs narratives. As we can see in Table 3, the models explore the use of CNNs, RNNs, QRNNs, different variants of attention mechanisms, and CRF; mixed models with two or more combinations were also explored. The dot product attention is the scaled version proposed by (Vaswani et al., 2017); the general attention is also known as Luong attention (Luong et al., 2015); the additive attention is also known as Bahdanau attention (Bahdanau et al., 2015).

Hyperparameters	Values
Conv. filters	35, 50, 100, 200
Kernel size	1, 3, 5, 7
Conv. dropout	0.0, 0.25, 0.5, 0.75
Recurrent hidden size	35, 50, 100, 200
Recurrent type	RNN, GRU, LSTM, QRNN
Recurrent dropout	0.0, 0.5
Attention dropout	0.0, 0.25, 0.5, 0.75
Attention variant	Dot Product, General, Additive
Attention hidden size	35, 50, 100, 200, 300
Number of heads	1, 2, 4
Multi-head hidden size	50, 100, 200

Table 3: Hyperparameters tried during greedy search.

We trained our models for a maximum of 40 epochs using small batch sizes. We found that smaller batch sizes work better in practice for models that have a CRF at the end, therefore we set the batch size to 1 for all configurations in order to have comparable results. We employed early stopping with patience of 5 epochs. We optimized the model's parameters using the Adam optimizer with the weight decay fix implementation (Loshchilov and Hutter, 2019). In order to avoid overfitting, we also used ℓ_2 regularization with $\lambda = 0.01$. For all other optimizers hyperparameters, we used the default ones defined on the PyTorch implementation. Finally, in all our experiments we used pre-trained word embeddings from (Treviso et al., 2017a), selecting the 600D Word2vec-skipgram type (due to its higher performance) and keeping them freezed during training.

Taking into account all configurations, more than 200 new architectures were trained and evaluated on the MCI class set of the Cinderella dataset. Since in this work we have more training data from different language tests, our main aim was that we can have a new model that performs better than the RCNN and can also generalize better to datasets of different story topics.

Due to space constraints we do not report the results for each configuration. Nevertheless, these experiments showed that dropout after convolutional and recurrent lay-

Model	Retelling			Sequenced Figures			Average
	MCIs	Controls	ADs	MCIs	Controls	ADs	
1. CRF	0.38	0.44	0.36	0.63	0.68	0.76	0.54
2. QRCNN	0.71	0.76	0.64	0.88	0.84	0.85	0.78
3. RCNN (original)	0.72	0.76	0.65	0.88	0.85	**0.91**	0.79
4. RCNN + CRF	0.74	0.76	0.65	0.88	0.85	0.85	0.79
5. CNN	0.72	0.76	0.65	0.88	0.85	0.85	0.79
6. CNN + CRF	0.72	0.76	0.63	0.88	0.85	0.83	0.78
7. CNN + ATTN	0.72	0.75	0.65	0.88	0.83	0.86	0.78
8. CNN + ATTN + CRF	0.74	0.76	0.65	**0.89**	0.84	0.83	0.79
9. RNN	0.73	0.76	0.64	**0.89**	0.85	0.88	0.79
10. RNN + CRF	0.76	0.78	0.66	**0.89**	**0.86**	0.85	0.80
11. RNN + ATTN	0.71	0.74	0.64	0.88	0.85	0.85	0.78
12. RNN + ATTN + CRF	**0.77**	**0.79**	**0.67**	**0.89**	**0.86**	0.85	**0.81**

Table 4: Cross-validation F_1 scores for each method on both retelling and sequenced figures datasets.

ers is an important factor to prevent overfitting. The dropout rate after convolutional layers was usually set to 0.25, and 0.5 for recurrent layers. For all models with convolutional layers, we found that the best kernel size was 7, and the best number of filters varied between 100 and 200. In general, the number of recurrent units varied between 100 a 200, except for models based on QRNNs, which performed better with 50 units. As for the recurrent unit type, we found that LSTMs usually perform better than GRUs and QRNNs. Finally, the general and additive attention variants were the ones that yielded the best results.

In order into train the best architectures with the datasets presented in Table 2, datasets of tasks that elicit narratives with visual stimuli (Cinderella and Dog Story) were joined and used for training the selected architectures. The datasets of tasks that elicit narratives with oral stimuli (Lucia and Wallet Story) were also joined and used for training the same architectures.

Taking in account the previous experiment with the MCI class set of The Cinderella Story dataset, we selected 12 different models (including the original RCNN) with their best configuration. Moreover, in order to show the impact of each architecture, we chose models that have unique configuration of layers. Table 4 shows 10-fold cross-validation results for datasets based on retellings and based on sequenced figures using these selected models.

With the exception of the CRF model, all others yield similar results, ranging from 0.78 to 0.81 F_1 score in average, with the best value being the RNN + ATTN + CRF model. Therefore, our answer to the question “Would new architectures have better performance?” is no, there was no significant increasing in F_1 score. It is a fact that more data taken from similar distribution (same task of a language battery) is beneficial. The RNN + ATTN + CRF model also makes a small improvement on the F_1 score when compared to the original RCNN model; however, the RCNN is still very competitive. Finally, it is worth noticing that all architectures use the same word embeddings extracted from (Treviso et al., 2017a), which were pre-trained using a large collection of written texts and, as a consequence, some of the lexical clues for sentence boundaries of spontaneous speech were probably not captured by this representation.

5. Exp. II: Does the Task Require a Specific Sentence Segmentation Detector?

An important question is whether models trained in a same task also generalize well to the other task being evaluated, i.e., can we use a model trained on retellings to segment narratives based on sequenced figures and vice versa?

To answer these questions, we chose 3 top recurrent models (9, 10 and 12) from the cross-validation experiments of Table 4 with addition of the original RCNN.

To avoid an unfair comparison between the datasets, the models were then trained again via 10-fold cross-validation where we also randomly split the other dataset into 10 folds to be used for testing. In order to evaluate each model, we first train the model using the training set of each fold and evaluated it twice, once using its test set counterpart and the other time using the respective fold of the other dataset. For example, if we trained the model for the 1st fold of sequenced figures, we tested it both on (i) its test subset; (ii) the 1st fold of the retelling dataset. Since we do not have a validation set, we can not employ an early stopping procedure, so instead we estimate a good number of epochs based on the average number of epochs obtained in the experiments reported on the Table 4.

Table 5 presents F_1 scores of our four best models from the previous experiment trained on the retelling datasets, and Table 6 presents F_1 scores of our four best models from the previous experiment trained on the sequenced figures datasets. In both tables the models were tested on both retelling and sequenced figures datasets.

Table 6 shows the best generalization result: the model RNN + ATTN + CRF. It was trained on sequenced-figures datasets and presents the best average values in both testing data (retellings and sequenced figures as well). The model RNN, in Table 6, also presents similar behavior. From these results, we can assume that sequenced-figures narratives bring linguistic features also present in retellings, but the reverse direction is not true, as we can see in Table 5. However, if a researcher will only work on retelling tasks, Table 5 shows that using only retelling datasets for training led to

Model	Retelling				Sequenced Figures				Average
	MCIs	Controls	ADs	Average	MCIs	Controls	ADs	Average	
RCNN (original)	0.84	0.80	0.75	0.80	0.51	0.60	0.44	0.52	0.66
RNN	0.72	0.77	0.61	0.70	**0.66**	**0.74**	**0.60**	**0.67**	**0.68**
RNN + CRF	0.84	**0.83**	**0.81**	**0.83**	0.51	0.57	0.41	0.50	0.66
RNN + ATTN + CRF	**0.85**	**0.83**	0.74	0.81	0.51	0.58	0.41	0.50	0.65

Table 5: F_1 scores of our best models trained on the retelling datasets and tested on both datasets.

Model	Sequenced Figures				Retelling				Average
	MCIs	Controls	ADs	Average	MCIs	Controls	ADs	Average	
RCNN (original)	0.68	0.74	0.61	0.68	0.67	0.70	**0.67**	**0.68**	0.68
RNN	0.69	0.74	**0.64**	0.69	**0.69**	0.71	0.63	**0.68**	0.68
RNN + CRF	**0.73**	**0.76**	0.61	**0.70**	0.67	0.71	0.61	0.66	0.68
RNN + ATTN + CRF	0.70	**0.76**	0.63	**0.70**	**0.69**	**0.72**	0.62	0.67	**0.69**

Table 6: F_1 scores of our best models trained on the sequenced figures datasets and tested on both datasets.

better results for the retelling task.

6. Exp. III: Does the Story Topic Demand a Specific Segmentation Detector?

Here, we evaluate if we can generalize on the topic of stories used in language batteries, allowing the creation of a generic and unique model for the sentence segmentation task for impaired speech transcriptions. Table 7 shows the results of our best models presented in Table 5 and 6 trained here with three datasets and tested with the fourth remaining dataset, totaling 16 models, and allowing a rich combination to evaluate the best results for the segmentation task. We show in bold the three best average values of F_1 scores (models 8, 9 and 10) (Avg 1). We also calculated the average of F_1 by model (RCNN, RNN, RNN+CRF and RNN+ATTN+CRF) and training class (same, all and MCI and Control classes) (Avg. 2), allowing us to find the model with the best generalization for new data, independently of the task.

Model 8 (RNN + ATTN + CRF) uses the datasets Cinderella, Dog and Wallet for training and Lucia for testing. The training was done with two datasets of narratives elicited by visual stimuli (Cinderella and Dog Story), which have already been selected as the best stimuli to generalize the best model in Experiment 2, taking in account the training with sequenced figures datasets (Table 6).

Considering a new dataset, independently of its task, the best model is the RNN, trained with "All groups" (Avg. 2). Comparing its value of F_1 (0.66) with the results of Table 1 (our previous generic segmentation system), there was a increase of 0.7 in the F_1 score.

Taking all these results into consideration, we chose the model RNN + ATTN + CRF for creating detectors for a specific task, i.e. training with datasets of tasks that elicit narratives with visual stimuli (Cinderella and Dog Story) and with datasets of tasks that elicit narratives with oral stimuli (Lucia and Wallet Story). Also, we chose to use only the groups of Controls and MCIs as their narratives are more similar than those of DAs. The RNN + ATTN + CRF model returned the best results in Experiments 1 and 2. Models with CRF generally do not generate two boundaries in sequence, as in "Ela saiu de. casa.", since the constraint of two periods in sequence is very strong. Moreover, by inspecting the predictions of these models we saw that the mean length of the sentences in the predicted transcriptions are very close to the ones in the training datasets (difference less than 1 most of the time). Although its number of parameters is slightly higher than that of the model RNN, the use of attention and CRF end up helping in the quality of the transcriptions. By looking at the results for Experiments 2 and 3, we also chose the RNN model, trained with Cinderella + Dog + Lucia datasets and with all classes, to be used as a unique and generic sentence segmentation detector due to its generalization performance.

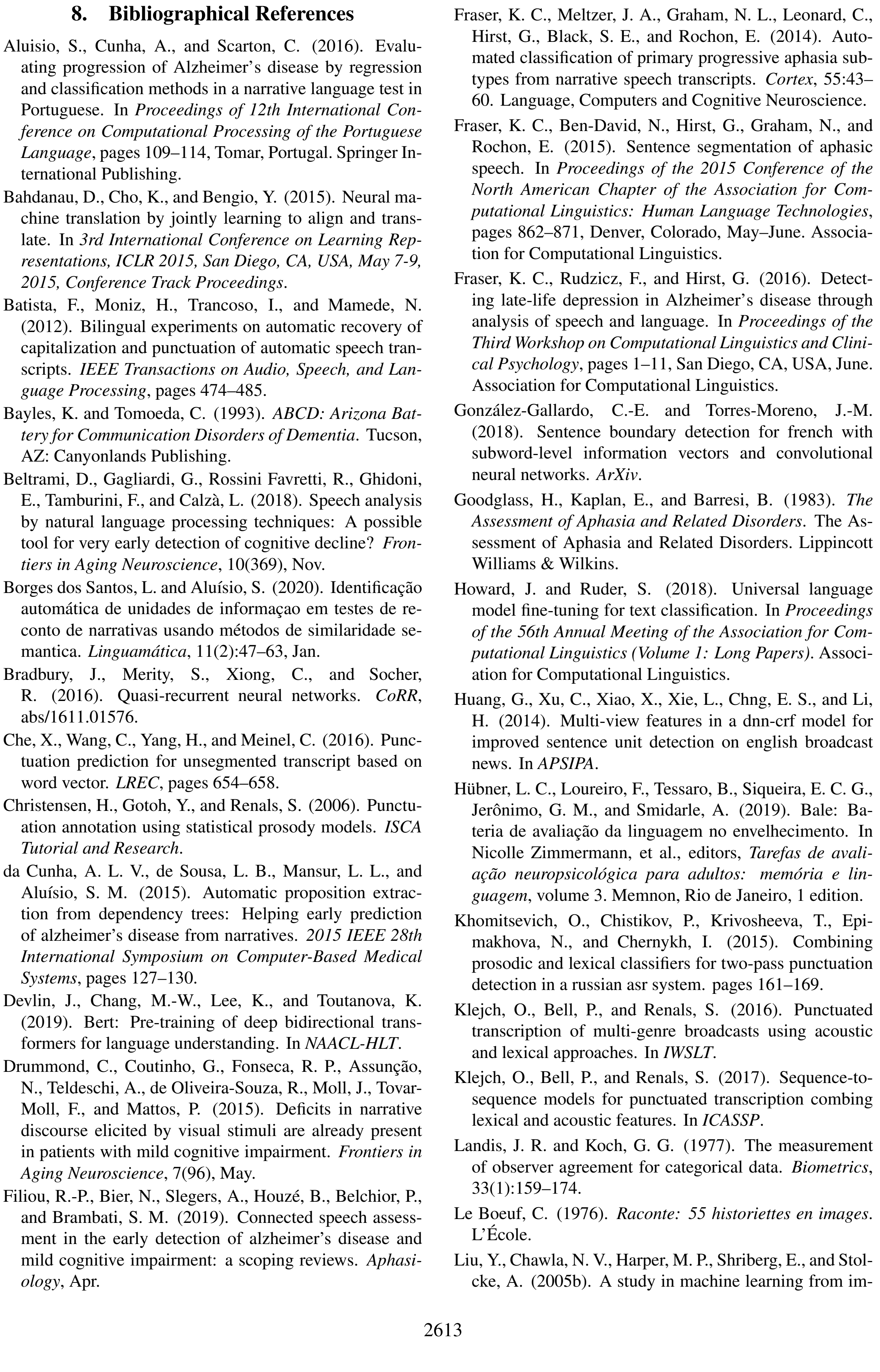
7. Conclusions and Future Work

In this paper, our main goal was to develop a robust and generic sentence segmentation system for narratives of language tests, based on experiments using four datasets of narratives used to evaluate cognitive processes. Instead, our study allowed us to develop and choose two new models — RNN and RNN + ATTN + CRF — for segmenting impaired speech transcriptions, along with an ideal combination of datasets and specific groups of narratives to be used as the training set. We chose the RNN + ATTN + CRF model for creating a segmentation detector for a specific task, because it returned the best results in Experiments 1 and 2. By analyzing the results from the Experiment 3, we chose the RNN for creating our generic segmentation detector. These findings are consistent with the model selected as the one that generalizes better for a different stimuli in Experiments 2 and 3. We also made publicly available the four datasets used in this study. Although we got better segmentations by applying Viterbi decoding for all of our models with CRF on top, the input for the Viterbi algorithm is the entire transcription, and therefore a global optimization over the sequence is being done, which might not be helpful at the end because there are several valid combina-

Model	Training datasets	Test dataset	Training class	MCIs	Controls	ADs	Avg. 1	Avg. 2
1. RCNN	Cinderella + Lucia + Wallet	Dog	Same class	0.49	0.59	0.61	0.56	0.60
			All classes	0.59	0.66	0.60	0.62	**0.65**
			MCIs and Controls	0.55	0.55	0.56	0.55	**0.65**
2. RNN	Cinderella + Lucia + Wallet	Dog	Same class	0.58	0.66	0.61	0.61	0.61
			All classes	0.62	0.65	0.59	0.62	**0.66**
			MCIs and Controls	0.55	0.64	0.57	0.59	0.64
3. RNN + CRF	Cinderella + Lucia + Wallet	Dog	Same class	0.54	0.56	0.56	0.56	0.58
			All classes	0.62	0.65	0.60	0.62	0.63
			MCIs and Controls	0.58	0.65	0.54	0.59	**0.64**
4. RNN + ATTN + CRF	Cinderella + Lucia + Wallet	Dog	Same class	0.53	0.57	0.50	0.53	0.56
			All classes	0.61	0.65	0.63	0.63	**0.65**
			MCIs and Controls	0.44	0.56	0.44	0.48	0.63
5. RCNN	Cinderella + Dog + Wallet	Lucia	Same class	n/a	0.71	0.56	0.63	
			All classes	n/a	0.75	0.69	0.72	
			MCIs and Controls	n/a	0.74	0.69	0.72	
6. RNN	Cinderella + Dog + Wallet	Lucia	Same class	n/a	0.71	0.56	0.64	
			All classes	n/a	0.72	0.62	0.67	
			MCIs and Controls	n/a	0.71	0.70	0.71	
7. RNN + CRF	Cinderella + Dog + Wallet	Lucia	Same class	n/a	0.76	0.59	0.67	
			All classes	n/a	0.72	0.63	0.68	
			MCIs and Controls	n/a	0.75	0.69	0.72	
8. RNN + ATTN + CRF	Cinderella + Dog + Wallet	Lucia	Same class	n/a	0.74	0.51	0.62	
			All classes	n/a	0.76	0.62	0.69	
			MCIs and Controls	n/a	0.77	0.71	**0.74**	
9. RCNN	Cinderella + Dog + Lucia	Wallet	Same class	0.63	0.71	n/a	0.67	
			All classes	0.70	0.74	n/a	0.72	
			MCIs and Controls	0.73	0.76	n/a	**0.74**	
10. RNN	Cinderella + Dog + Lucia	Wallet	Same class	0.64	0.70	n/a	0.67	
			All classes	0.75	0.76	n/a	**0.75**	
			MCIs and Controls	0.71	0.71	n/a	0.71	
11. RNN + CRF	Cinderella + Dog + Lucia	Wallet	Same class	0.55	0.60	n/a	0.58	
			All classes	0.69	0.65	n/a	0.67	
			MCIs and Controls	0.70	0.67	n/a	0.69	
12. RNN + ATTN + CRF	Cinderella + Dog + Lucia	Wallet	Same class	0.61	0.63	n/a	0.62	
			All classes	0.72	0.67	n/a	0.69	
			MCIs and Controls	0.73	0.70	n/a	0.72	
13. RCNN	Wallet + Dog + Lucia	Cinderella	Same class	0.57	0.60	0.49	0.55	
			All classes	0.57	0.59	0.52	0.56	
			MCIs and Controls	0.60	0.61	0.52	0.58	
14. RNN	Wallet + Dog + Lucia	Cinderella	Same class	0.54	0.59	0.47	0.54	
			All classes	0.60	0.61	0.53	0.58	
			MCIs and Controls	0.60	0.61	0.51	0.57	
15. RNN + CRF	Wallet + Dog + Lucia	Cinderella	Same class	0.56	0.60	0.36	0.51	
			All classes	0.59	0.61	0.47	0.55	
			MCIs and Controls	0.61	0.63	0.50	0.58	
16. RNN + ATTN + CRF	Wallet + Dog + Lucia	Cinderella	Same class	0.53	0.60	0.26	0.46	
			All classes	0.60	0.61	0.51	0.57	
			MCIs and Controls	0.61	0.62	0.51	0.58	

Table 7: F_1 scores of the 16 models trained on a combination of three datasets and tested on the fourth remaining dataset. “Same class” means that the method was trained only with the same specific class used for testing. “All classes” means that data from all classes were used for training. Avg. 1 means average of F_1 in a row; Avg. 2 means average of F_1 by model and training class, averaging over all the combinations of training/test datasets.

tions of boundary and non-boundary states. Thus, a local decoding strategy, like posterior decoding, might be more helpful in this scenario. Another direction is to follow the recent trend of the NLP community and encode our input using contextual representations from pretrained language models like ELMo and BERT (Peters et al., 2018; Devlin et al., 2019), yet a fine-tunning procedure on a large dataset of spontaneous speech transcriptions is still probably needed (Howard and Ruder, 2018). Finally, we plan to do evaluations with the output of an ASR system, as a high word recognition error rate can greatly affect our results.



8. Bibliographical References

Aluisio, S., Cunha, A., and Scarton, C. (2016). Evaluating progression of Alzheimer's disease by regression and classification methods in a narrative language test in Portuguese. In *Proceedings of 12th International Conference on Computational Processing of the Portuguese Language*, pages 109–114, Tomar, Portugal. Springer International Publishing.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Batista, F., Moniz, H., Trancoso, I., and Mamede, N. (2012). Bilingual experiments on automatic recovery of capitalization and punctuation of automatic speech transcripts. *IEEE Transactions on Audio, Speech, and Language Processing*, pages 474–485.

Bayles, K. and Tomoeda, C. (1993). *ABCD: Arizona Battery for Communication Disorders of Dementia*. Tucson, AZ: Canyonlands Publishing.

Beltrami, D., Gagliardi, G., Rossini Favretti, R., Ghidoni, E., Tamburini, F., and Calzà, L. (2018). Speech analysis by natural language processing techniques: A possible tool for very early detection of cognitive decline? *Frontiers in Aging Neuroscience*, 10(369), Nov.

Borges dos Santos, L. and Aluísio, S. (2020). Identificação automática de unidades de informaçao em testes de reconto de narrativas usando métodos de similaridade semantica. *Linguamática*, 11(2):47–63, Jan.

Bradbury, J., Merity, S., Xiong, C., and Socher, R. (2016). Quasi-recurrent neural networks. *CoRR*, abs/1611.01576.

Che, X., Wang, C., Yang, H., and Meinel, C. (2016). Punctuation prediction for unsegmented transcript based on word vector. *LREC*, pages 654–658.

Christensen, H., Gotoh, Y., and Renals, S. (2006). Punctuation annotation using statistical prosody models. *ISCA Tutorial and Research*.

da Cunha, A. L. V., de Sousa, L. B., Mansur, L. L., and Aluísio, S. M. (2015). Automatic proposition extraction from dependency trees: Helping early prediction of alzheimer's disease from narratives. *2015 IEEE 28th International Symposium on Computer-Based Medical Systems*, pages 127–130.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Drummond, C., Coutinho, G., Fonseca, R. P., Assunção, N., Teldeschi, A., de Oliveira-Souza, R., Moll, J., Tovar-Moll, F., and Mattos, P. (2015). Deficits in narrative discourse elicited by visual stimuli are already present in patients with mild cognitive impairment. *Frontiers in Aging Neuroscience*, 7(96), May.

Filiou, R.-P., Bier, N., Slegers, A., Houzé, B., Belchior, P., and Brambati, S. M. (2019). Connected speech assessment in the early detection of alzheimer's disease and mild cognitive impairment: a scoping reviews. *Aphasiology*, Apr.

Fraser, K. C., Meltzer, J. A., Graham, N. L., Leonard, C., Hirst, G., Black, S. E., and Rochon, E. (2014). Automated classification of primary progressive aphasia subtypes from narrative speech transcripts. *Cortex*, 55:43–60. Language, Computers and Cognitive Neuroscience.

Fraser, K. C., Ben-David, N., Hirst, G., Graham, N., and Rochon, E. (2015). Sentence segmentation of aphasic speech. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 862–871, Denver, Colorado, May–June. Association for Computational Linguistics.

Fraser, K. C., Rudzicz, F., and Hirst, G. (2016). Detecting late-life depression in Alzheimer's disease through analysis of speech and language. In *Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology*, pages 1–11, San Diego, CA, USA, June. Association for Computational Linguistics.

González-Gallardo, C.-E. and Torres-Moreno, J.-M. (2018). Sentence boundary detection for french with subword-level information vectors and convolutional neural networks. *ArXiv*.

Goodglass, H., Kaplan, E., and Barresi, B. (1983). *The Assessment of Aphasia and Related Disorders*. The Assessment of Aphasia and Related Disorders. Lippincott Williams & Wilkins.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Huang, G., Xu, C., Xiao, X., Xie, L., Chng, E. S., and Li, H. (2014). Multi-view features in a dnn-crf model for improved sentence unit detection on english broadcast news. In *APSIPA*.

Hübner, L. C., Loureiro, F., Tessaro, B., Siqueira, E. C. G., Jerônimo, G. M., and Smidarle, A. (2019). Bale: Bateria de avaliação da linguagem no envelhecimento. In Nicolle Zimmermann, et al., editors, *Tarefas de avaliação neuropsicológica para adultos: memória e linguagem*, volume 3. Memnon, Rio de Janeiro, 1 edition.

Khomitsevich, O., Chistikov, P., Krivosheeva, T., Epimakhova, N., and Chernykh, I. (2015). Combining prosodic and lexical classifiers for two-pass punctuation detection in a russian asr system. pages 161–169.

Klejch, O., Bell, P., and Renals, S. (2016). Punctuated transcription of multi-genre broadcasts using acoustic and lexical approaches. In *IWSLT*.

Klejch, O., Bell, P., and Renals, S. (2017). Sequence-to-sequence models for punctuated transcription combing lexical and acoustic features. In *ICASSP*.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Le Boeuf, C. (1976). *Raconte: 55 historiettes en images*. L'École.

Liu, Y., Chawla, N. V., Harper, M. P., Shriberg, E., and Stolcke, A. (2005b). A study in machine learning from imbalanced data for sentence boundary detection in speech. *Computer Speech and Language*.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Luong, T., Pham, H., and Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, September. Association for Computational Linguistics.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL*.

Shriberg, E., Stolcke, A., Hakkani-Tür, D., and Tür, G. (2000). Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, pages 127–154.

Shriberg, E., Favre, B., Fung, J., Hakkani-tür, D., and Cuendet, S. (2009). Prosodic similarities of dialog act boundaries across speaking styles. *Linguistic Patterns in Spontaneous Speech*, pages 213–239.

Tilk, O. and Alumäe, T. (2016). Bidirectional recurrent neural network with attention mechanism for punctuation restoration. In *Interspeech*, pages 3047–3051.

Tillas, A. (2015). Language as grist to the mill of cognition. *Cognitive Processing*, 16(3):219–243, Aug.

Toledo, C. M., Aluísio, S. M., Dos Santos, L. B., Brucki, S., Trés, E. S., d. O. M. O., and Mansur, L. L. (2017). Analysis of macrolinguistic aspects of narratives from individuals with alzheimer's disease, mild cognitive impairment, and no cognitive impairment. *Alzheimer's dementia (Amsterdam, Netherlands)*, 10.

Treviso, M., Shulby, C., and Aluísio, S. (2017a). Evaluating word embeddings for sentence boundary detection in speech transcripts. In *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, pages 151–160, Uberlândia, Brazil, October. Sociedade Brasileira de Computação.

Treviso, M., Shulby, C., and Aluísio, S. (2017b). Sentence segmentation in narrative transcripts from neuropsychological tests using recurrent convolutional neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 315–325, Valencia, Spain, April. Association for Computational Linguistics.

Treviso, M. V., dos Santos, L. B., Shulby, C., Hübner, L. C., Mansur, L. L., and Aluísio, S. M. (2018). Detecting mild cognitive impairment in narratives in brazilian portuguese: first steps towards a fully automated system. *Letras de Hoje*, 53(1):48–58, jan.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wechsler, D. (1997). *Wechsler Memory Scale - Third Edition.* The Psychological Corporation, San Antonio, TX.

Yancheva, M., Fraser, K., and Rudzicz, F. (2015). Using linguistic features longitudinally to predict clinical scores for Alzheimer's disease and related dementias. In *Proceedings of SLPAT 2015: 6th Workshop on Speech and Language Processing for Assistive Technologies*, pages 134–139, Dresden, Germany, September. Association for Computational Linguistics.

A.2 Deep Learning against COVID-19: Respiratory Insufficiency Detection in Brazilian Portuguese Speech

Título:	**Deep Learning against COVID-19: Respiratory Insufficiency Detection in Brazilian Portuguese Speech**
Autores:	**Edresson Casanova, Lucas Gris, Augusto Camargo, Daniel da Silva, Murilo Gazzola, Ester Sabino, Anna Levin, Arnaldo Candido Jr, Sandra Aluisio e Marcelo Finger**
Ano:	**2021**
Conferência:	**Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021**
Situação:	**Publicado**

Contribuições relevantes do artigo:

- Mostra que é possível detectar a insuficiência respiratória analisando enunciados falados em condições da vida real, utilizando aprendizado profundo;
- Propõe uma nova arquitetura para detecção de insuficiência respiratória analisando enunciados de fala;
- Apresenta um método para lidar com áudios do mundo real, tornando possível a analise de enunciado falados em condições da vida real;
- Disponibiliza publicamente um dataset para detecção de insuficiência respiratória em fala de pacientes com COVID-19.

Deep Learning against COVID-19: Respiratory Insufficiency Detection in Brazilian Portuguese Speech

Edresson Casanova[1], Lucas Gris[2], Augusto Camargo[3], Daniel da Silva[2], Murilo Gazzola[1], Ester Sabino[4], Anna S. Levin[4], Arnaldo Candido Jr[2], Sandra Aluisio[1], Marcelo Finger[3]†

[1]Instituto de Ciências Matemáticas e de Computação, University of São Paulo, São Carlos/SP, Brazil
[2]Federal University of Technology – Paraná, Medianeira/PR, Brazil
[3]Instituto de Matemática e Estatítica, University of São Paulo, São Paulo/SP, Brazil
[4]Faculdade de Medicina, University of São Paulo, São Paulo/SP, Brazil

Abstract

Respiratory insufficiency is a symptom that requires hospitalization. This work investigates whether it is possible to detect this condition by analyzing patient's speech samples; the analysis was performed on data collected during the first wave of the COVID-19 pandemic in 2020, and thus limited to respiratory insufficiency in COVID-19 patients. For that, a dataset was created consisting of speech emissions of both COVID-19 patients affected by respiratory insufficiency and a control group. This dataset was used to build a Convolution Neural Network to detect respiratory insufficiency using speech emission MFCC representations. Methodologically, dealing with background noise was a challenge, so we also collected background noise from COVID-19 wards where patients were located. Due to the difficulty in filtering noise without eliminating crucial information, noise samples were injected in the control group data to prevent bias. Moreover, we investigated (i) two approaches to address the duration variance of audios, and (ii) the ideal number of noise samples to inject in both patients and the control group to prevent bias and overfitting. The techniques developed reached 91.66% accuracy. Thus we validated the project's Leading Hypothesis, namely that it is possible to detect respiratory insufficiency in speech utterances, under real-life environmental conditions; we believe our results justify further enquiries into the use of automated speech analysis to support health professionals in triage procedures.

1 Introduction

This work started as part of the academic initiative to help in the effort to deal with the COVID-19 pandemic in a severely affected region in Brazil. COVID-19 is an infectious disease caused by the virus SARS-CoV-2. This illness is mainly associated to severe acute respiratory syndrome, although it is harmful to other organs, like heart, kidney and brain. About 82% of cases are mild or moderate, while the rest are severe or grave, demanding hospitalization or intensive care. The most vulnerable groups are people over the 60's, and people with specific medical conditions such as diabetes, obesity, hypertension and heart disease. According to WHO[1], in August 3 2020, more than 19.2 million people in the world had contracted COVID-19, with a Case Fatality Ratio of CFR=2.8%. Respiratory Insufficiency (RI) is a symptom that requires hospitalization, which is aggravated due to a frequent COVID-19 condition called *silent hypoxia*, low blood oxygen concentration without breath shortness (Tobin et al., 2020).

This work ***leading hypothesis*** states that *it is possible to detect respiratory insufficiency by analyzing spoken utterances in real-life conditions*, typically a moderately large sentence, thus subscribing to the view of *speech as a biomarker*. This work aims at validating this leading hypothesis using deep learning techniques.

If the hypothesis holds, it will motivate further enquiries on the use of automated speech analysis to support health professionals; with infectious diseases such as COVID-19, a serious concern involves deciding whether an RI suspect should stay in social isolation or be directed to a medical facility. Project SPIRA[2][3] was initiated to investigate the feasibility of supporting medical triage of patients with COVID-19 symptoms by remotely detecting respiratory insufficiency through automated speech utterance analysis, where no other resources are

[†] Corresponding author: mfinger@ime.usp.br .

[1] https://covid19.who.int, visited May 24 2021.

[2] https://spira.ime.usp.br/

[3] In Portuguese, ***S****istema de detecção* ***P****recoce de* ***I****nsuficiência* ***R****espiratória por análise de* ***A****udio* – system for early detection of respiratory insufficiency via audio analysis.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 625–633
August 1–6, 2021. ©2021 Association for Computational Linguistics

available other than a phone line or a cellphone app. A positive result may motivate further research into speech-based remote detection of respiratory problems originating from other causes, such as heart condition, airway obstruction, severe asthma, H1N1, etc.

This research started as a response to the peak of the first COVID-19 wave in 2020, when health infrastructure was overloaded, so no doctors nor nurses were available for data collection, and there was no triage point available for research. Thus, COVID-19 patient utterances were collected mostly by medical students at COVID-19 wards from patients with blood oxygenation below 92%, as an indication of respiratory insufficiency, and control data was collected by voice donations over the internet, assumed healthy, with no access to blood oxygenation. Recordings were made in out-of-studio conditions, using portable recording equipment employed in noisy wards. On the other hand, conditions for healthy voice donations over the Internet and using diverse sound equipment had a large variation. This *audio data in-the-wild* approach was assumed from the start as part of the challenge of validating the leading hypothesis. Part of the methodological novelty of this work lies on how to deal with these conditions. This task required a multidisciplinary group involving medical doctors, linguists, speech therapists and computer scientists, all of which were aware of those conditions and challenges facing us.

This work proposes a machine learning method to detect respiratory insufficiency by analyzing voice audio recordings of sentences long enough to feature respiratory pauses in speech. The test is very cheap, requiring only a voice sample from each patient and maybe employed where no other medical equipment is available.In order to tackle the audio analysis, we propose the use of deep artificial neural networks over Mel Frequency Cepstral Coefficients (MFCCs) (Logan et al., 2000) extracted from patient's audios.

The code and datasets are publicly available at `https://github.com/SPIRA-COVID19/SPIRA-ACL2021`, under a CC BY-SA 4.0 license.

This paper is organized as follows: Section 2 discusses related work. In Section 3, the dataset acquired, the preprocessing steps, the noise insertion procedure, the proposed model and experiments are described, respectively. Afterwards, the models obtained are evaluated and discussed in Section 4. Finally, Section 5 presents the conclusions and final thoughts.

2 Related work

COVID-19 is a recent disease. However, even before the eruption of the pandemic, we could already find in the literature a few explorations of speech as a biomarker (Botelho et al., 2019; Trancoso et al., 2019; Nevler et al., 2019), with some recent recommendations (Robin et al., 2020).

Several initiatives can be found on the Web that record human voice in order to assess the presence and the gravity of COVID-19, e.g. the COVID-19 Sounds data collection initiative (Brown et al., 2020) and startup initiative aiming to develop a pre-diagnostic tool[4]. Those works aim to diagnose COVID-19 from voice or breathing or coughing sounds, and there are some initial positive results on COVID-19 detection in asymptomatic individuals (Laguarta et al., 2020). Unlike our approach, no work aimed specifically at respiratory insufficiency or at patient triage, but they propose to employ some form of artificial intelligence processing.

In similarity to our goals, there have been recent proposals of applications for the triage of patients using natural language processing of texts extracted from radiology reports (Hassanpour et al., 2017) and patient questionnaires (Spasić et al., 2019). So language, both as text and now as speech, is being used for patient screening.

Moreover, Neural Networks and Convolutional Neural Networks (CNNs) have been used in noisy environments mostly, but not exclusively, for fault diagnosis (Zhang et al., 2018; Munir et al., 2019), noise reduction in voice processing (Maas et al., 2012) and medical ECG diagnosis (Acharya et al., 2017). On the other hand, *noise injection* was a technique used in the past to avoid overfitting in training Neural Networks (Matsuoka, 1992; Grandvalet et al., 1997; Zur et al., 2009), as opposed to avoiding classification biases, as in our approach.

3 Methodology

In order to build a neural network model for the proposed task, it is necessary to gather a dataset containing voices of healthy individuals and COVID-19 patients (Section 3.1). The resulting dataset required several preprocessing treatments and noise treatment, as discussed in Sections 3.2 and 3.3. The next step was to propose several neural models to

[4] https://www.voicemed.io/

investigate the best one for the task (Section 3.4) evaluated according to experiments carried over the dataset (Section 3.5).

3.1 Dataset

The dataset creation was composed of two parts and an "appendix". The first part consisted of audios gathered via Web by a system specifically designed for this task[5], from May to July of 2020. Healthy volunteers were asked to donate audio samples via a web interface. This allowed us to build our control group. In order to do that, the system URL was disclosed through local news and social networking. The resulting dataset part is composed, after elimination of blank samples, of more than 6 thousands voice donors. No blood oxygen saturation information was available for the control group.

In the second part, we collected audios from patients infected by SARS-CoV-2 from June to July of 2020. This collection was performed in COVID-19 wards in two university hospitals, in São Paulo city, Brazil, restricted to patients with blood oxygenation level (SpO2) inferior to 92%, as an indication of respiratory insufficiency. This allowed us to collect 536 samples from patients in different age groups. Several problems led to discarding patient voice samples, chiefly among which were collectors whispering during collection; a large set of collection instructions was assembled during the period in which voice collection took place. It is important to note that São Paulo is a local and international hub, with a large migrant and immigrant population. Hospitals received COVID-19 patients from the city as well as from adjoining regions. Collection was absolutely anonymous, so no one knows who were the patients and controls, and no ethnographic information is available. On the other hand, this allowed us to release the data.

As a COVID-19 ward is a noisy environment, an "appendix" was built for this dataset, consisting of samples of pure background noise at the ward (no voice), typically collected at the start of a collection session. This is an important piece of information, as the ward noise is very different from the background noise found in the control group, and consists of a *data bias* that has to be controlled during experiments.

The gathered audios contain three utterances:

- Utterance 1, a moderately long sentence containing 31 syllables, designed by linguists to allow for spontaneous breathing breaks, while being relatively simple to be spoken, even by low literacy voice donors: *"O amor ao próximo ajuda a enfrentar o coronavírus com a força que a gente precisa."* ("Love of neighbor helps in strengthening the fight against Coronavirus.");

- Utterance 2, a well known nursery rhyme for donors having reading difficulties, due to lack or reading glasses in hospital, or other types of reading impediments: *"Batatinha quando nasce, espalha a rama pelo chão, nenezinho quando dorme põe a mão ao coração"* ("When small potatoes germinate, branches sprout on the ground; when baby sleeps, hands rest over the heart");

- Utterance 3, a widely known song, on the lines of "Happy birthday to you": *"Parabéns a você, nesta data querida, muitas felicidades, muitos anos de vida"* ("Happy birthday to you, on this dear date, lots of happiness, many years of life").

Collecting longer utterances was totally impractical in a COVID-19 ward. The collection had to be adapted to what was possible in that context.

We identified several issues with the original dataset that need to be addressed. First, there is class imbalance, as we have fewer positive instances (COVID-19 patients) than negative ones (healthy individuals from the control group). Second, it is sex imbalanced, as a greater number of healthy women participated in the process than healthy men. Additionally, there are more men in COVID-19 wards than women. Third, there is an age imbalance, as there are more elderly in hospital care than young people in our observations. Fourth, we also detected utterance imbalance, as utterance 1 was more common among patients; healthy people typically recorded all proposed utterances. Fifth, the control group presented popping and crackling noise, possible due to the characteristics from the recording devices. Furthermore, as mentioned above, wards tend to be noisy environments.

We addressed most of the dataset issues by sample balancing, taking advantage of the greater number of control group samples. Only audios from utterance 1 were selected and the number of samples used in experiments was balanced by class and sex, but not by age, to avoid drastically reduc-

[5] https://spira.ime.usp.br/coleta/

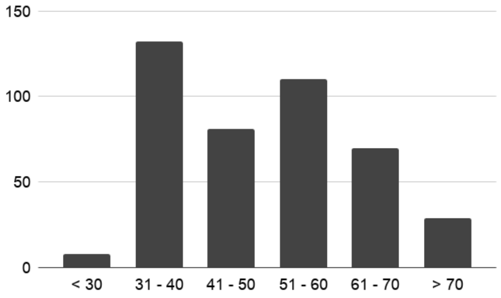


Figure 1: Distribution of Ages in the Dataset

ing the available data. Overall age distribution is presented in Figure 1.

We also had to discard audio containing the collector's (whispering) voice. The most serious issue for bias removal, though, is the presence of ward background noise in patient audios; we observed that it is easier to insert ward noise in the control group than to remove it from the patients' signal. This process will be addressed in Section 3.2.

The dataset was divided in training, validation and test, as is usual in statistical learning. We selected audios with the best signal-noise ratio to use in the test set, and the second best audios were used for validation. The aim of this partitioning is to detect training overfitting.

Information of the resulting filtered dataset is presented in Table 1.

3.2 Pre-processing

In general, the majority of the audios in the dataset was sampled at 48kHz. We pre-processed these files using Torch Audio 0.5.0 in the following way. First, for dimensionality reduction reasons, we re-sampled these audios at 16kHz. Second, we extracted the MFCCs using a 400ms window employing Fast Fourier Transform (FFT) (Brigham and Morrow, 1967), with hop length 160 and 1,200 FFT components, of which we retained only 40 coefficients. Before the MFCC feature extraction process though, we need to address the difference of duration present in our data.

The duration of our samples in the dataset varies, in which audios from the positive class are slightly longer than audios from the negative class, as presented in Table 1. We have developed two approaches to deal with this phenomenon. First, we applied padding in the instances during training. This is equivalent to complete the audios with silence so that all audios have the same duration. Second, we have extracted fixed length fragments from the audios. This approach aims to prevent the model from performing the classification giving too much importance to the audio length. In order to augment the training data, windowing with 1 second steps was applied to extract audio fragments.

3.3 Noise Insertion During Training

Ward noise is a serious bias source, as confirmed by our preliminary experiments (Section 4). In this scenario, a neural network can be biased during training by focusing only on background noise. One possible alternative would be noise filtering, but besides the possibility of inserting extra biases due to differential noise suppression in patient and control audio samples, there is also the possibility of suppressing important low-energy information that allows for the distinction between healthy and respiratory affected speech samples.

To address this issue, we decided to record pure background noise samples from COVID-19 wards and to inject into patients and control group audios. In total, 16 samples with approximately 1 minute each were recorded.

The inserted noise can also be a cause of bias, as the model can extract specific features from the noise recording. To avoid this kind of bias, we decided to inject noise in all samples. We had the option of inserting in training, validation and testing samples, which will be described in Section 4. We can also control the amount of noise samples inserted in each audio.

In our experiments, we investigate the ideal number of noise samples to inject in both patients and the control group. This had a big impact in overfitting prevention, as a form of unbiased learning, as described in Section 4. During training, at each epoch, audio samples can be injected with one or more distinct noise samples. Each time a given audio is used for training, noise samples are drawn from the noise base. Besides that, the start point of each noise sample is also randomized. Finally, we also draw a factor to change the intensity of the sample. This factor is constrained by a maximum amplitude value, which was determined from the analysis of patient audio noises. The aim is to insert noises as similar as possible to already pre-existing noise. We also executed the same experiment three times with different random seeds to obtain better measures of the noise insertion impact.

Table 1: Filtered dataset information

Sets	Control			Patients			Total Audios	Total Duration (s)
	Male	Female	Mean Duration (s)	Male	Female	Mean Duration (s)		
Training	59	84	8.15	83	66	13.18	292	3110
Validation	8	8	7.75	8	8	10.78	32	296
Test	22	26	7.77	28	32	9.43	108	983

The test and validation sets were created in such a way to allow overfitting detection as they are composed mostly of audios with very limited amount of noise. As a result, we cannot apply k-fold Cross Validation and similar methods. We compensate this by running the same experiment three times with different random seeds. This fact, together with the dynamic noise insertion during training, allows us to obtain averaged accuracy for each experiment.

3.4 Proposed Model

Several models were tested in preliminary experiments and we describe the one that led to the best results.

This process involved three main aspects: (a) the topology and model parameters; (b) the main hyper-parameters; (c) regularization. The last is especially important, since our dataset contains several issues that can lead to overfitting.

Regarding topology and model parameters, preliminary experimental results showed that CNNs applied to MFCCs are useful to analyze this kind of problem. Other preliminary experiments investigated spectrograms and topologies like fully-connected and recurrent networks, which showed lower performance than the chosen topology. Figure 2 presents the chosen model's main features including layers, filters, kernels, number of neurons and activation functions. The following conventions are adopted in the figure: kernel size is represented by K; convolutional dilation size (Yu and Koltun, 2015) is represented by D; and fully connected layers are represented by FC. The input size is omitted because these parameters changed according to the experiment and will be detailed in Section 3.5. We investigated the use of Mish activation function (Misra, 2019), due to its regularization effects during training, which helps prevent overfitting.

Regarding the main hyper-parameters, we have used the Binary Cross-Entropy as loss, and Adam optimizer (Kingma and Ba, 2014). The initial learning rate was set to 10^{-3}, and the Noam's decay scheme (Vaswani et al., 2017) was applied on each 1,000 steps. For each experiment presented in Section 3.5, we trained the model for 1,000 epochs using a batch size of 30.

Regarding regularization, overfitting mitigation is a major concern given our dataset noise characteristics. Therefore, several approaches for regularization were applied. Besides Mish as an activation function, we used three other strategies. First, a global weight decay of 0.01 was applied. Second, a dropout of 0.70 was used in all layers, except in the output layer. Last, we applied group normalization (Wu and He, 2018) after each convolutional layer. The group normalization was applied on pairs of convolution filters. Therefore, the number of groups is half the number of filters.

3.5 Experiments

For the experiments we explored three main aspects with respect to noise insertion and duration variance: (a) overfitting impact; (b) padding vs windowing approach (using four second windows or adding padding); and (c) the ideal number of noise samples. Table 2 presents the proposed experiments and their results.

First we investigated if the model can overfit when trained over original audios (experiments 1.x). In this series of experiments, we trained the model using both approaches of duration variance.

Second we analyzed two approaches to address the duration variance: audio padded to the maximum length of the dataset; windowing using the aproach described in Section 3.2 (experiments 2.x). Specifically, we presented padding application only in experiments 1.1 and 2.1 because experiments showed that the windowed approach led to more robust results. When padding is used, the accuracy is calculated as usual. However, in windowed experiments, several audio fragments are extracted and their predictions averaged for the classification decision. Regarding window size, we have chosen four seconds, considering our smallest audio in the

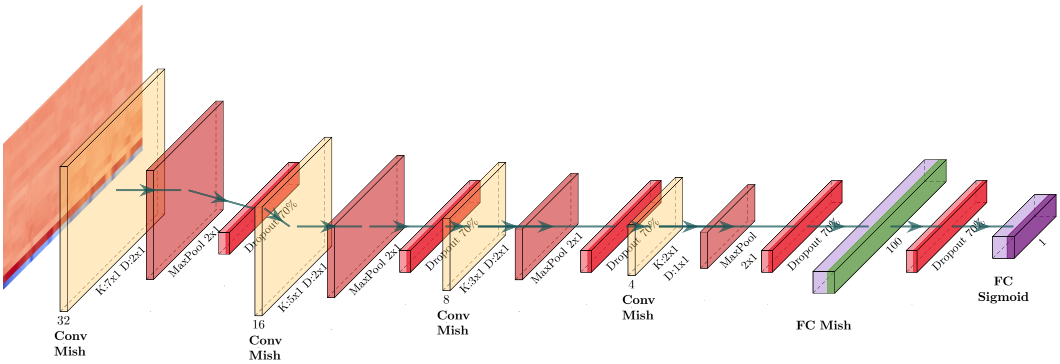


Figure 2: CNN topology proposed with four convolutional layers and two fully connected layers

Table 2: Proposed experiments and results

Description	Exp.	Duration Approach	Noise Samples		Accuracy (without noise in test samples)	Accuracy (with noise in test samples)	Training time (h)
			Patient	Control			
Overfitting Analysis	1.1	Padding	0	0	98.15 ± 0.93	50.93 ± 0.53	4.25
	1.2	Windowing	0	0	98.15 ± 0.53	50.93 ± 0.93	9.07
Duration Variance Analysis	2.1	Padding	0	1	61.11 ± 8.40	74.07 ± 1.93	4.77
	2.2	Windowing	0	1	66.67 ± 3.74	86.11 ± 2.98	9.58
Noise Insertion Analysis	3.1	Windowing	1	1	80.56 ± 2.45	68.52 ± 1.41	6.57
	3.2	Windowing	1	2	84.26 ± 6.17	83.33 ± 3.34	12.27
	3.3	Windowing	2	2	88.89 ± 0.53	85.19 ± 0.93	13.00
	3.4	Windowing	2	3	74.07 ± 5.10	85.19 ± 1.85	13.67
	3.5	Windowing	3	3	91.67 ± 2.98	87.04 ± 0.93	14.70
	3.6	Windowing	3	4	62.96 ± 8.35	74.07 ± 2.45	11.85
	3.7	Windowing	4	4	88.89 ± 1.41	83.33 ± 1.07	10.40
	3.8	Windowing	4	5	56.48 ± 5.10	72.22 ± 9.99	9.83
	3.9	Windowing	5	5	70.37 ± 15.8	69.44 ± 9.27	10.55
	3.10	Windowing	5	6	51.85 ± 3.51	61.11 ± 2.98	11.18
	3.11	Windowing	6	6	74.07 ± 10.7	74.07 ± 8.83	11.98
	3.12	Windowing	6	7	50.00 ± 0.53	54.63 ± 3.51	12.63

dataset contains 4.6 seconds and new data samples can be even smaller.

Third we examine the ideal number of noise samples to be inserted to prevent overfitting (experiments 3.x), using the best duration approach according to experiments 2.x. For each experiment, we tested the model using both noise insertion and no noise insertion to analyze performance.

Our model was implemented using Pytorch 1.5.1. We ran the experiments on a NVIDIA Titan V GPU with 12GB RAM in a server with Intel(R) Core(TM) i7-8700 CPU and 16GB of RAM.

4 Results and Discussion

To better understand bias and overfitting we used a test set containing only audios with a minimal amount of noise. The accuracy of each experiment is presented in Table 2, both with and without artificial insertion of ward noise in test samples.

Experiments 1.x showed the model is biased without noise insertion in the training set. We note a high accuracy in experiments 1.1 and 1.2 without noise in training and testing; in contrast, when noise is inserted in all test samples, it classifies all samples as coming from patients. We interpret this as a strong indication that the model is biased by the presence of noise in the patient samples.

Experiments 2.x showed that windowing (2.2) is preferable over padding (2.1), as described in Section 3; the model performs better when the windowed approach is used, that is, 66% using windowing against 61% using padding. We consider this as evidence of susceptibility to bias by padding. In fact, padding inserts a considerable amount of silence, specially in patient samples, and the windowed approach works as a data augmentation technique, as more instances are generated in this process.

Experiment 3.x were used to determine the optimal amount of noise insertion. Note that sometimes better results were obtained without noise in test samples and sometimes the other way around. In general, the bias is greatly reduced by inserting at least one noise sample on the negative instances. As expected, the insertion of too much noise decreases the model performance. The best overall accuracy was obtained in experiment 3.5, which reached 91% accuracy in the task. For experiment 3.5, we obtained F1 = 0.90, without noise insertion; with noise insertion, F1 = 0.87.

Figure 3 presents the loss variation of the best model (experiment 3.5) during training. Early stopping is used to get the best iterations after approximately 20k steps.

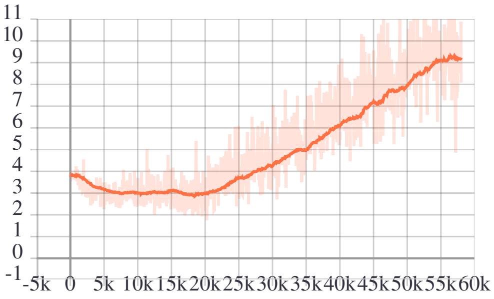


Figure 3: Validation Loss for Experiment 3.5 during Training

Figure 4 shows the model performance over the number of noise samples inserted. With respect to the number of noise samples, our experiments suggest that a number of 2 to 4 noise insertions in each audio provides best accuracy. In each case, two possibilities have been tested, namely the insertion of an equal number of noise samples in each training audio, and the insertion of one extra noise sample to control audio, assuming that patient audios already have the original ward noise. It was initially expected that the insertion of an extra noise sample in control audios would produce better results; surprisingly, the opposite effect was observed. The possible explanation for this observation is that there are times when wards are calmer and silent and the insertion of noise in control audios leads to bias. This is especially true for testing samples, due to the criteria used to build the testing set.

5 Conclusions and future work

In the effort to tackle the COVID-19, we have developed a method to classify real-life speech audio signals on whether or not that signal originated from a person suffering from respiratory insufficiency. In this effort, we obtained 91.67% accuracy, thus validating the hypothesis that such a detection is feasible, and that human speech can be treated as a biomarker in this case.

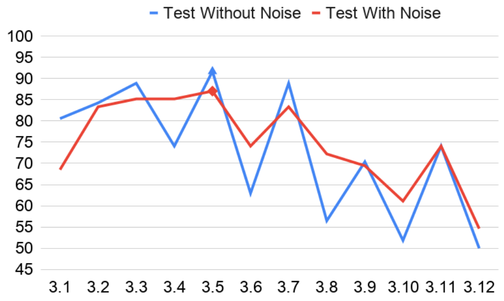


Figure 4: Sample Noise Analysis for the Best Experiments

One important consequence of this work was the construction of a dataset containing voice samples of COVID-19 patients with respiratory insufficiency and also a set of samples of environmental noise, which were central in treating real-life sound samples. Noise insertion was chosen as the more adequate option when contemplating the biases that would be incurred by filtering procedures. In particular, it made sense to add ward noise to the existing ward samples as a way to balance the biases that were incurred by the necessary addition of ward noise to control data. In this way, all data (patient and control) suffered from similar manipulation, avoiding editing bias, and experiments showed that such a procedure produced best results. This aimed at preventing the models from memorizing ward noise and editing distortion information instead of COVID-19 features.

There was a considerable difficulty to collect voice data from infected patients during the pandemic. The size of the patient dataset reflects the limitations on collections in COVID-19 wards. Moreover, the use of audio from different environments was absolutely unavoidable, as we only had access to patients in COVID-19 wards, where no control subjects were available. Therefore, control data had to be collected in a different environment. As a result, the amount of data was scarce, and data augmentation techniques were designed for such a setting; our results indicate that it was not an excessive amount of data augmentation, as consistent results were obtained over a large variety of experiments. We hope that with the weakening of the emergency situation, it could become easier to collect data from patients with respiratory insufficiency.

Future work includes augmenting the dataset with audios collected at the triage point, whether in hospital admission rooms, or through a remote admission system. In this way, speech audio signals from both sufferers and non-sufferers of respiratory insufficiency would be obtained under similar conditions. This would allow us to extend this study to other respiratory illnesses besides COVID-19. Also, other neural architectures can be explored, as well as smarter feature engineering.

Acknowledgments

This work was supported by Fapesp project 2020/06443-5 (SPIRA). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

Marcelo Finger was partly supported by Fapesp projects 2019/07665-4 and 2014/12236-1 and CNPq grant PQ 303609/2018-4.

We also gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPU used in part of the experiments presented in this research.

References

U. Rajendra Acharya, Hamido Fujita, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, and Muhammad Adam. 2017. Application of deep convolutional neural network for automated detection of myocardial infarction using ecg signals. *Information Sciences*, 415-416:190 – 198.

M Catarina Botelho, Isabel Trancoso, Alberto Abad, and Teresa Paiva. 2019. Speech as a biomarker for obstructive sleep apnea detection. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5851–5855. IEEE.

E Oran Brigham and RE Morrow. 1967. The fast fourier transform. *IEEE spectrum*, 4(12):63–70.

Chloë Brown, Jagmohan Chauhan, Andreas Grammenos, Jing Han, Apinan Hasthanasombat, Dimitris Spathis, Tong Xia, Pietro Cicuta, and Cecilia Mascolo. 2020. Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data. *arXiv preprint arXiv:2006.05919*.

Yves Grandvalet, Stéphane Canu, and Stéphane Boucheron. 1997. Noise injection: Theoretical prospects. *Neural Computation*, 9(5):1093–1108.

Saeed Hassanpour, Curtis Langlotz, Timothy Amrhein, Nicholas Befera, and Matthew Lungren. 2017. Performance of a machine learning classifier of knee mri reports in two large academic radiology practices: A tool to estimate diagnostic yield. *AJR. American Journal of Roentgenology*, 208:1–4.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

J. Laguarta, F. Hueto, and B. Subirana. 2020. Covid-19 artificial intelligence diagnosis using only cough recordings. *IEEE Open Journal of Engineering in Medicine and Biology*, 1:275–281.

Beth Logan et al. 2000. Mel frequency cepstral coefficients for music modeling. In *Ismir*, volume 270, pages 1–11.

Andrew Maas, Quoc V. Le, Tyler M. O'Neil, Oriol Vinyals, Patrick Nguyen, and Andrew Y. Ng. 2012. Recurrent neural networks for noise reduction in robust asr. In *INTERSPEECH*.

K. Matsuoka. 1992. Noise injection into inputs in back-propagation learning. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3):436–440.

Diganta Misra. 2019. Mish: A self regularized non-monotonic neural activation function. *arXiv preprint arXiv:1908.08681*.

Nauman Munir, Hak-Joon Kim, Jinhyun Park, Sung-Jin Song, and Sung-Sik Kang. 2019. Convolutional neural network for ultrasonic weldment flaw classification in noisy conditions. *Ultrasonics*, 94:74 – 81.

Naomi Nevler, Sharon Ash, David J Irwin, Mark Liberman, and Murray Grossman. 2019. Validated automatic speech biomarkers in primary progressive aphasia. *Annals of Clinical and Translational Neurology*, 6(1):4–14.

J. Robin, J. E. Harrison, L. D. Kaufman, F. Rudzicz, W. Simpson, and M.J. Yancheva. 2020. Evaluation of speech-based digital biomarkers: Review and recommendations. *Digital Biomarkers*, 4(3):99–108.

I. Spasić, D. Owen, A. Smith, and K. Button. 2019. Klosure: Closing in on open–ended patient questionnaires with text mining. *Journal of Biomedical Semantics*, 10.

Martin J Tobin, Franco Laghi, and Amal Jubran. 2020. Why covid-19 silent hypoxemia is baffling to physicians. *American Journal of Respiratory and Critical Care Medicine*, 202(3):356–360.

Isabel Trancoso, Maria Joana Ribeiro Folgado Correia, Francisco Teixeira, Alberto Abad, Maria Catarina Tavares Botelho, and Bhiksha Raj. 2019.

Speech as a (private?) biomarker for speech affecting diseases. In *In ICIEA 2019 - The 14th IEEE Conference on Industrial Electronics and Applications*, Xi'an, China. IEEE. Keynote paper.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Yuxin Wu and Kaiming He. 2018. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19.

Fisher Yu and Vladlen Koltun. 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Wei Zhang, Chuanhao Li, Gaoliang Peng, Yuanhang Chen, and Zhujun Zhang. 2018. A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load. *Mechanical Systems and Signal Processing*, 100:439 – 453.

Richard M Zur, Yulei Jiang, Lorenzo L Pesce, and Karen Drukker. 2009. Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical physics*, 36(10):4810–4818.

A.3 Transfer Learning and Data Augmentation Techniques to the COVID-19 Identification Tasks in ComParE 2021

Título:	**Transfer Learning and Data Augmentation Techniques to the COVID-19 Identification Tasks in ComParE 2021**
Autores:	**Edresson Casanova, Arnaldo Candido Jr., Ricardo Corso Fernandes Jr., Marcelo Finger, Lucas Rafael Stefanel Gris, Moacir A. Ponti, Daniel Peixoto Pinto da Silva**
Ano:	**2021**
Conferência:	**INTERSPEECH 2021**
Situação:	**Publicado**

Contribuições relevantes do artigo:

- Explora o uso de redes neurais artificiais pré-treinadas com áudios em larga escala para a identificação de COVID-19 em amostras de fala e tosse utilizando um dataset multilíngue;
- Melhora a generalização dos modelos profundos explorados por meio das técnicas de aumento de dados Mixup (ZHANG *et al.*, 2018), SpecAugment (PARK *et al.*, 2019) e ruído aditivo (SNYDER *et al.*, 2018).

Transfer Learning and Data Augmentation Techniques to the COVID-19 Identification Tasks in ComParE 2021

Edresson Casanova[1], *Arnaldo Candido Jr.*[2], *Ricardo Corso Fernandes Jr.*[2], *Marcelo Finger*[3], *Lucas Rafael Stefanel Gris*[2], *Moacir A. Ponti*[1], *Daniel Peixoto Pinto da Silva*[2]

[1] Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos/SP, Brazil
[2] Federal University of Technology – Paraná, Medianeira, PR, Brazil
[3] Dept. of Computer Science, Institute of Mathematics and Statistics, University of São Paulo

edresson@usp.br

Abstract

In this work, we propose several techniques to address data scarceness in ComParE 2021 COVID-19 identification tasks for the application of deep models such as Convolutional Neural Networks. Data is initially preprocessed into spectrogram or MFCC-gram formats. After preprocessing, we combine three different data augmentation techniques to be applied in model training. Then we employ transfer learning techniques from pretrained audio neural networks. Those techniques are applied to several distinct neural architectures. For COVID-19 identification in speech segments, we obtained competitive results. On the other hand, in the identification task based on cough data, we succeeded in producing a noticeable improvement on existing baselines, reaching 75.9% unweighted average recall (UAR).

Index Terms: computational paralinguistics, COVID-19, deep learning, data augmentation, transfer learning

1. Introduction

Automated analysis of audio signals turned out to be a promising path for the screening of respiratory diseases [1, 2]. One year after COVID-19 was officially declared a global pandemic, there is a huge interest on improving such tools, allowing alternative forms of identification of suspicious cases of infection.

The Interspeech Computational Paralinguistics ChallengE (ComParE) 2021 proposes two competitions related to detecting COVID-19 from audio samples. Such samples represent speech and cough from both healthy and infected speakers. The COVID-19 Speech Sub-Challenge (CSS) offers 3.24 hours of audio recordings containing speech samples, while the COVID-19 Cough Sub-Challenge (CCS) provides 1.63 hours of cough samples.

Our contribution to ComParE 2021 explores four architectures based on convolutional neural networks aiming at detecting COVID-19 in both CSS and CCS. In particular, we investigate employing transfer learning, a well-established technique to address data scarceness and adaptation between different datasets [3], which has been used successfully in previous editions of ComParE [4, 5, 6, 7]. Thus our proposal studies transfer learning from Pretrained Audio Neural Networks (PANNs) [8], which are models trained on millions of audio samples. In addition, we explore three different data augmentation techniques. The main contributions can be summarized as: (i) investigating large-scale pretrained audio neural networks for the identification of COVID-19; (ii) improving the generalization of deep models via three data augmentation techniques, i.e. Mixup [9], SpecAugment [10] and additive noise data augmentation [11].

2. Experimental Framework

First, we present the original datasets (section 2.1), noting that there are insufficient data for a deep learning approach. Due to this scarcity, we applied three techniques for data augmentation: Noise Data Augmentation (section 2.2), SpecAugment (section 2.3) and Mixup (section 2.4), in that order and combining the three methods. Finally, we also explore transfer learning to address data scarcity (section 2.5).

2.1. Datasets

The datasets for the sub-challenges were extracted from the COVID-19 Sound database [12, 13, 14]. Two groups were defined, audio recordings from subjects with COVID-19 (patient group) and from individuals not infected and therefore without COVID-19 symptoms (control group).

Regarding the CSS, 893 audio clips were gathered from 366 speakers and different languages. There are 3.24 hours of recordings, with duration varying between 3.6 and 30.1 seconds. A total of 315 instances for training: 243 from the control group and 72 for patient group. A development set is also available with 295 samples: 153 for control and 142 for patients. The test set consists of 283 audios without public labels [14].

For the CCS, 929 cough audio instances were obtained from 397 speakers, resulting in 1.63 hours. The training set presents 215 and 71 audios for control and patients, respectively (286 in total). The development set contains 231 audios (183 for control and 48 for patients). The test set is composed of 208 unlabeled audio clips [14].

Both the CSS and CCS audios are sampled at 16khz. It is known the dataset contains multiple instances per speaker. In addition, the training, development, and test sets do not share speakers, thus they are speaker independent [14].

2.2. Noise data augmentation

We use the additive noise method which is popular in speech processing [11]. For this purpose, we use the MUSAN corpus [15], which is composed of three subsets: the noise subset, which contains approximately 6 hours of noise, such as ambient noise and dialtones; the music subset, which contains approximately 42 hours of music and; the speech subset, containing 60 hours of human speech. We choose to use only the noise subset, because it contains ambient noise that represents most of the noise found in the datasets used in this paper. A noise sample in every training step is chosen randomly and is added to the original signal with a signal to noise ratio random varying from 0 to 15dB, similar to the proposal of [16].

2.3. SpecAugment

SpecAugment [10] is an augmentation method originally proposed for automatic speech recognition. It consists of masking a log-mel spectrogram in the frequency and time domain [8]. The frequency masking is applied so that f consecutive mel channels $[f0, f0 + f]$ are masked, where f is chosen from a uniform distribution ranging from zero to a previously defined value that indicates the maximum masking width. Finally, $f0$ is chosen at random from $[0, v - f]$, where v is the number of mel channels [10]. Masking in time domain is similar to frequency masking, but along the time axis [8]. Similar to [8], we used two masks in the frequency domain, with a maximum width of eight mel channels and two masks in the time domain with a maximum width of 64 frames.

2.4. Mixup

Mixup is a technique proposed by [9] to address overfitting and sensitivity, using adversarial examples in deep neural networks. The authors showed that this method improves the generalization of deep models. In this method, at each training step, the neural network is trained on a convex combination of two inputs and their corresponding labels [6], creating interpolations between pairs of instances. Given two inputs x_i and x_j and their corresponding classes y_i and y_j, their resulting Mixup $(\widetilde{x}, \widetilde{y})$ is defined in Eq. (1).

$$\begin{aligned} \widetilde{x} &= \lambda x_i + (1 - \lambda) x_j \\ \widetilde{y} &= \lambda y_i + (1 - \lambda) y_j, \end{aligned} \tag{1}$$

The parameter λ in (1) is sampled for each pair (i, j) from a Beta Distribution with parameters α, $\lambda \sim Beta(\alpha, \alpha)$, where $\alpha \in (0, \infty)$ [9]. We tested several possible Beta distributions with $\alpha \in (0, 1)$, following guidelines of [6].

2.5. PANNs: AudioSet Pre-training

Transfer learning has shown to be a promising technique in several tasks [17, 18, 19], such as computer vision [18], where pretrained models are used as feature extractors in images [20], videos [21], and show to be useful when different datasets are used as source and target tasks [3]. Finally, in the audio domain, transfer learning obtained from images is also used in audio pattern recognition tasks [4]. The Pretrained Audio Neural Networks (PANNs) [8] were proposed in order to support several tasks such as audio tagging, acoustic scene classification, music classification, speech emotion classification and sound event detection.

PANNs used in this were trained in AudioSet [22], a dataset containing approximately 1.9 million audio samples from 527 different classes. The authors explored different convolutional topologies, such as VGG-like CNNs [23], MobileNets [24] and ResNets [25]. As a result, the authors reported the best models, which were trained using as input log-mel spectrograms extracted from audios sampled at 32 kHz. The log-mel spectrogram was extracted using Fast Fourier Transform (FFT) [26], with Hamming window 1024, hop length 320 and 1024 FFT components. In addition, 64 mel filter banks were applied to calculate the log-mel spectrogram. Overall, there is good empirical evidence in favor of applying PANNs in audio pattern recognition tasks [8], especially under limited availability of training data. In this work we explore transfer learning from three PANNs, namely CNN14, ResNet-38 and MobileNetv1.

3. Experiments

In order to build a classification model for COVID-19 detection, four scenarios of data augmentation and neural network topologies were investigated. Firstly, we present the four experiments in section 3.1. Then, detailed explanation regarding implementation is given in section 3.2. Finally, we describe the choice of hyper-parameters used to build each model in section 3.3.

3.1. Proposed Experiments

Each experiment is based on a distinct neural network topology and investigates the use of transfer learning and different data augmentation approaches. To facilitate reproducibility, the source code used in each experiment is available on GitHub[1]. The experiments were carried out as follows:

- **SpiraNet:** uses the SPIRA-project topology proposed specifically for detecting respiratory insufficiency in speech audio samples represented as Mel Frequency Cepstral Coefficients (MFCCs) [27]. The authors proposed a convolutional neural network to identify respiratory insufficiency in infected patients with COVID-19 in a scenario of high environmental noise, achieving promising results (∼ 91% accuracy). In this experiment, we use the same topology. Given the dataset difference, we searched for alternative strategies and fine-tuned hyper-parameters, namely the kernel size, convolutional dilatation, dropout, number of fully connected layer neurons after the last convolutional layer (FC dim), learning rate, weight decay and optimizer for each of the dataset (CSS, CCS). We use MFCCs with the same parameters reported in [27]. For this experiment we *do not* use noise data augmentation, Mixup and SpecAugment.
- **CNN14:** this experiment explores transfer learning from the PANN CNN14 model [8]. We adjusted the following original hyper-parameters in order to increase performance in the CSS and CCS datasets: Mixup α, learning rate, weight decay and optimizer for each of the datasets. The input is composed of log-mel spectrograms. We used the same parameters as the PANNs to extract the log-mel spectrogram (described in Section 2.5), except for the sampling rate, which was 16 kHz as in CSS and CCS datasets. This models makes use of noise data augmentation, Mixup and SpecAugment.
- **ResNet-38:** it is similar to CNN14, but exploring transfer learning of the PANN ResNet-38 [8].
- **MobileNetv1:** also similar to CNN14, but exploring transfer learning from PANN MobileNetv1 [8].

Each proposed experiment was executed using three different configurations: simple holdout, windowed holdout and K-fold cross validation. In all the cases, classifier ensembles were created.

Regarding the simple holdout method, the official training and development sets were used. This method was applied in two stages. First, the hyper-parameters of the four neural networks were fine-tuned, one at a time, using a fixed random seed. In the second phase, each experiment was carried out five times using five different random seeds and an ensemble was created, performing vote by sum of probabilities. We report only the results for the second phase.

[1] `https://github.com/Edresson/SPIRA-ComParE2021`

For the windowed holdout, the procedure is similar to the simple holdout, however we do not use the entire sample of audios for training the networks. In this approach, training is performed by selecting a random window from the original input audio, the window size varies according to the dataset used (detailed in Section 3.2). For development and test, the classifiers received all possible windows from the audio considering the hop size as window size (no overlapping) and their results are combined in the voting process.

Finally, in the k-fold cross validation, we merged the training and development sets, using five folds to evaluate each experiment. In this approach, each fold is used in the training of ensembles rather than single models. Each ensemble is composed of the five models (obtained with five different random seeds initialization), yielding 25 models over all folds. The development set results are obtained analysing each fold individually, by voting. In the test set, however, we perform a different approach, inspired in [4], joining all 25 models to vote. This approach has the main advantage of covering more samples from the training and development sets in order to decide the class. As a drawback, the speaker independence assumption presented in Section 2.1 may be compromised. Despite this, in [4] the authors showed that this approach was effective in a scenario without the independence of speakers between the train and development set, turning out to be the winner of the ComParE 2020 Mask Sub-Challenge.

3.2. Experiment Implementation Details

In all experiments, our models are trained for 100 epochs. We use the Binary Cross Entropy function [28] as loss for training experiments based on Mixup. On the other hand, for experiments that do not use Mixup, the loss function used was the average of the Binary Cross Entropy calculated individually for each class. Finally, in all experiments we choose the best checkpoint using the average of the Binary Cross Entropy calculated individually for each class in the development set.

The audio duration was set as the maximum duration on the dataset for simple holdout and cross validation experiments in order to standardize the audio duration, allowing network training and inference. Zero padding was used to adjust audios duration when needed, injecting silence in the samples. For windowed holdout, an audio window is chosen at random and the window size is three seconds for the CSS dataset and two seconds for the CCS dataset. These values were chosen because they represent the rounded down value of the smallest audios in each of the datasets.

Because the dataset has significant class imbalance, for the k-fold cross validation, each fold is generated using proportional stratified sampling. Batches, however, uses a different method, either for cross validation or other holdout approaches. Weighted random sampling [29] is used to build a batch containing a balanced number of instances for each class. In this technique, each instance receives a weight indicating its probability to be selected [30]. The inverse frequency of each class is used to defined the weights of its instances.

All of our models were implemented using Pytorch 1.6.0 [29] and trained on an NVIDIA Titan RTX GPU with 24GB of memory on a server with Intel (R) Core (TM) i9-10900 CPU and 128GB of RAM. In addition, all models were trained with a batch size of 16 and using Noam's decay scheme [31] applied to every 500 steps.

3.3. Models hyper-parameters

All the hyper-parameter values presented in this section were manually adjusted, one at a time for both simple and windowed holdout approaches. On the other hand, K-fold procedure uses the same hyper-parameter values as in holdout. For this adjusts, we used the development set loss.

Regarding **CSS**, the following hyper-parameters were chosen in both holdout and k-fold evaluation. Adam [32] was used for experiments 1, 3 and 4, while RAdamc [33] was used in experiment 2. The initial learning rate were defined as 0.1 for experiment 1 and 0.001 for the others. Weight decay of 0.01 was used in experiment 1 and 4, while 1e-05 was used for the others. Mixup $\alpha = 1$ were used in experiment 2, 3 and 4. Experiment 1 had other parameters adjusted, namely in FC dim 100, dropout rate 0.7, convolutional dilatation 2×1 and kernel size 5×1. The remaining experiments use transfer learning and these hyper-parameters are kept at the default values.

In the windowed holdout approach, we used Adam optimizer for experiment 3 and AdamW optimizer [34] for the others. The learning rate was set to 0.01 in experiment 1 and 0.001 for the others. Experiments 1 and 4 had weight decay of 0.0001, experiment 2 had the value set to 0.001 and experiment 3 used no weight decay. Mixup $\alpha = 0.8$, 0.3 and 0.9 where used in experiments 2, 3 and 4, respectively. Experiment 1 also had its hyper-parameters tuned with FC dim equal to 75, dropout rate of 0.7, convolutional dilatation of 2×1 and kernel sizes of 7×1, 5×1, 3×1 and 2×1 for each respectively layer.

Regarding **CCS**, the following same hyper-parameters were used for simple holdout and k-fold cross validation approaches. The choice of optimizers is the same for CSS, except experiment 2, which used AdamW. The initial learning rate were defined 0.01 for experiment 4 and 0.001 for the others. Weight decays of 0.1, 0.01, 1e-05 and 0 were used for experiments 1 to 4, respectively. Mixup α 0.9, 1.0 and 0.7 were used in experiment 2, 3 and 4. As in the CSS analysis, experiment 1 had some parameters adjusted, namely, FC dim 125, dropout rate 0.7, convolutional dilatation 2×1 and kernel sizes 2×1, 2×1, 2×1 and 5×1.

In the windowed holdout approach, we used Adam optimizer in all experiments. The learning rate was set to 0.01 in experiment 4 and 0.001 for the others. Experiments 1 and 4 had weight decay of 0.0001, experiment 2 had the value set to 0.001 and experiment 3 used no weight decay. Mixup α of 0.8, 0.3 and 0.9 where used in experiments 2, 3 and 4, respectively. Experiment 1 also had its hyper-parameters tuned, FC dim 75, dropout rate 0.7, convolutional dilatation 2×1 and kernel sizes 7×1, 5×1, 3×1 and 2×1. Weight decays were 0.01, 0.001, 1e-05 and 0 and experiments 1 to 4, respectively. Mixup $\alpha = 0.4$, 0.1, 0.4 were used for experiments 2 to 4. Finally, experiment 1 had extra hyper-parameters adjusted was FC dim 75, dropout rate 0.7, convolutional dilatation 2x1 and as kernel sizes 7×1, 5×1, 3×1 and 2×1.

4. Results and Discussion

Tables 1 and 2 presents the obtained results for CSS and CCS, respectively. For each approach, we selected the experiment with highest development UAR and used it for test evaluation. We also build two ensembles of ensembles (sum of probabilities), one including all experiments and other with the three highest development UARs.

Regarding **CSS**, CNN14 provided consistent results over all sampling methods for the development set, resulting in UARs

Table 1: *CSS experiments*

Exp.	**Train**		**Devel**		**Test**
	F1	UAR	F1	UAR	UAR
Baseline [14]	-	-	-	57.90	**72.10**
Simple Holdout					
SpiraNet	91.50	96.34	70.49	73.57	
CNN14	92.61	96.27	74.13	76.94	
ResNet-38	95.77	96.81	**76.40**	**78.39**	70.30
MobileNet	**97.95**	**99.38**	73.80	75.73	
Windowed Holdout					
SpiraNet	65.83	79.39	49.13	59.28	
CNN14	**90.32**	**95.93**	**76.25**	**76.08**	65.20
ResNet-38	84.56	90.86	72.65	75.00	
MobileNet	83.22	89.96	75.26	75.93	
Cross K-fold					
SpiraNet	(84.57)	(88.38)	(76.43)	(81.80)	
CNN14	**(99.82)**	**(99.90)**	**(87.37)**	**(90.59)**	68.90
ResNet-38	(99.65)	(99.81)	(86.34)	(89.75)	
MobileNet	(92.83)	(95.18)	(79.06)	(83.99)	
Heterogeneous Ensambles					
Baseline [14]	-	-	-	-	71.10
Top three	-	-	-	-	69.70
All Models	-	-	-	-	69.40

Table 2: *CCS experiments*

Exp.	**Train**		**Devel**		**Test**
	F1	UAR	F1	UAR	UAR
Baseline [14]	-	-	-	64.7	72.90
Simple Holdout					
SpiraNet	57.65	74.83	41.21	62.73	
CNN14	**91.61**	**96.97**	**51.42**	**69.92**	**75.90**
ResNet-38	85.02	94.18	47.76	68.57	
MobileNet	72.04	83.40	43.24	64.75	
Windowed Holdout					
SpiraNet	64.96	77.77	38.51	60.41	
CNN14	57.89	75.31	43.42	65.26	
ResNet-38	75.44	86.69	45.76	66.37	
MobileNet	**59.84**	**72.57**	**50.00**	**67.77**	68.90
Cross K-fold					
SpiraNet	(86.10)	(94.29)	(55.95)	(72.17)	
CNN14	**(99.79)**	**(99.93)**	**(76.79)**	**(86.60)**	69.60
ResNet-38	(99.79)	(99.93)	(70.75)	(82.48)	
MobileNet	(71.57)	(85.21)	(64.13)	(79.22)	
Heterogeneous Ensambles					
Baseline [14]	-	-	-	-	73.90
Top three	-	-	-	-	71.20
All Models	-	-	-	-	70.60

superior to 76%. In fact, this model lead to highest UARs in both windowed holdout and k-fold cross validation. The best UAR for simple holdout were obtained by the ResNet-38 model.

In [14], the authors reached a maximum UAR of 70.50% in the development set and this same model reached the second best result in the test with a UAR of 68.70%. Compared to this model, our best development model has a development UAR approximately 8% higher and a test UAR and 2% higher. However, a baseline with inferior development UAR leaded to the highest test UAR. Our model is approximately 2% inferior to this baseline. Additionally, our two heterogeneous ensembles had a similar result, but both failed to overcome the baseline in the test set.

The best model of the baseline in the test set was the second worst considering development set and jumped from 57.90% to 72.10% from one set to the other. This difference of approximately 14% may indicate that the development set is not a good representative of the test set. As it is a multi-language dataset, the unbalance of languages in the validation set can compromise learning, this information is unclear.

In **CCS** experiments, CNN14 also showed consistent results, presenting the highest UARs for simple holdout and k-fold cross validation sampling approaches and MobileNet presented the higher development UAR for the windowed sampling method.

In [14] the authors reached a maximum UAR of 66.40% in the development set and this same model reached a UAR of 67.60% in the test. Our best model in development shows superior results of approximately 3% when compared to the best model in the baseline development set.

In the test set, our best experiment using the simple holdout approach and the CNN14 architecture reached a UAR of 75.90%, which was 2% above the ensemble used as a baseline in the competition. In addition, we see baselines with superior performances in the test set compared to the performance in the development set. In [14], the best model was more than 8% better in test than in development. The same occurred with our best model it was approximately 6% better in test set. Finally, our two heterogeneous ensembles had results close to each other, however they did not surpass our best homogeneous ensemble performance.

The sampling methods showed a tendency to better results for simple holdout, followed by cross validation and windowed holdout, in this order. Despite cross validation presented promising results in related work [4], it was not able to surpass simple-holdout. One hypothesis is that this dataset is smaller to the one used in [4]. It is noticeable that k-fold was still superior to windowed holdout even without specific fine tunning as in the other two approaches.

5. Conclusions and future work

This paper presented a contribution for the CSS and CCS competitions by tackling the challenges using deep learning based models. Because such methods are data-hungry, in contrast with the size of the datasets available for the challenge, we explored both transfer learning and several data augmentation methods, in attempt to obtain competitive results. In this sense, we also explored instance sampling methods. Although our models could not overcome the baseline results for CSS, they were able to surpass them in CCS by 2%.

6. Acknowledgements

This work was supported by Fapesp project 2020/06443-5 (SPIRA). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Marcelo Finger was partly supported by Fapesp projects 2019/07665-4 and 2014/12236-1 and CNPq grant PQ 303609/2018-4. We also gratefully acknowledge the support of NVIDIA Corporation with the donation of the GPU used in part of the experiments presented in this research.

7. References

[1] G. Chambres, P. Hanna, and M. Desainte-Catherine, "Automatic detection of patient with respiratory diseases using lung sound analysis," in *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*. IEEE, 2018, pp. 1–6.

[2] D. Perna and A. Tagarelli, "Deep auscultation: Predicting respiratory anomalies and diseases via recurrent neural networks," in *2019 IEEE 32nd International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2019, pp. 50–55.

[3] F. P. Dos Santos, C. Zor, J. Kittler, and M. A. Ponti, "Learning image features with fewer labels using a semi-supervised deep convolutional network," *Neural Networks*, vol. 132, pp. 131–143, 2020.

[4] J. Szep and S. Hariri, "Paralinguistic classification of mask wearing by image classifiers and fusion," *Proceedings INTERSPEECH. Shanghai, China: ISCA*, pp. 2087–2091, 2020.

[5] M. Markitantov, D. Dresvyanskiy, D. Mamontov, H. Kaya, W. Minker, and A. Karpov, "Ensembling end-to-end deep models for computational paralinguistics tasks: Compare 2020 mask and breathing sub-challenges," *INTERSPEECH, Shanghai, China*, 2020.

[6] T. Koike, K. Qian, B. W. Schuller, and Y. Yamamoto, "Learning higher representations from pre-trained deep models with data augmentation for the compare 2020 challenge mask task," *Proceedings INTERSPEECH. Shanghai, China: ISCA*, pp. 2047–2051, 2020.

[7] S.-L. Yeh, G.-Y. Chao, B.-H. Su, Y.-L. Huang, M.-H. Lin, Y.-C. Tsai, Y.-W. Tai, Z.-C. Lu, C.-Y. Chen, T.-M. Tai *et al.*, "Using attention networks and adversarial augmentation for styrian dialect continuous sleepiness and baby sound recognition." in *INTERSPEECH*, 2019, pp. 2398–2402.

[8] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[9] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[10] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Proceedings INTERSPEECH 2019*, pp. 2613–2617, 2019.

[11] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[12] C. Brown, J. Chauhan, A. Grammenos, J. Han, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic diagnosis of covid-19 from crowdsourced respiratory sound data," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 3474–3484.

[13] J. Han, C. Brown, J. Chauhan, A. Grammenos, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, and C. Mascolo, "Exploring automatic covid-19 diagnosis via voice and symptoms from crowdsourced data," *arXiv preprint arXiv:2102.05225*, 2021.

[14] B. W. Schuller, A. Batliner, C. Bergler, C. Mascolo, J. Han, I. Lefter, H. Kaya, S. Amiriparian, A. Baird, L. Stappen, S. Ottl, M. Gerczuk, P. Tzirakis, C. Brown, J. Chauhan, A. Grammenos, A. Hasthanasombat, D. Spathis, T. Xia, P. Cicuta, M. R. Leon J. J. Zwerts, J. Treep, and C. Kaandorp, "The INTERSPEECH 2021 Computational Paralinguistics Challenge: COVID-19 Cough, COVID-19 Speech, Escalation & Primates," in *Proceedings INTERSPEECH 2021, 22nd Annual Conference of the International Speech Communication Association*. Brno, Czechia: ISCA, September 2021, to appear.

[15] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[16] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova baseline system for the voxceleb speaker recognition challenge 2020," *arXiv preprint arXiv:2009.14153*, 2020.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[18] S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2661–2671.

[19] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.

[20] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.

[21] F. P. dos Santos, L. S. Ribeiro, and M. A. Ponti, "Generalization of feature embeddings transferred from different video anomaly detection domains," *Journal of Visual Communication and Image Representation*, vol. 60, pp. 407–416, 2019.

[22] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[26] E. O. Brigham and R. Morrow, "The fast fourier transform," *IEEE spectrum*, vol. 4, no. 12, pp. 63–70, 1967.

[27] E. Casanova, L. Gris, A. Camargo, D. Silva, M. Gazzola, E. Sabino, A. Levin, A. Candido Jr, S. Aluisio, and M. Finger, "Deep learning against covid-19: Respiratory insufficiency detection in brazilian portuguese speech," in *Findings of the Association for Computational Linguistics: ACL 2021*. ACL, Aug. 2021, accepted for publication.

[28] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[29] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

[30] T. Emara, H. M. Afify, F. H. Ismail, and A. E. Hassanien, "A modified inception-v4 for imbalanced skin cancer classification dataset," in *2019 14th International Conference on Computer Engineering and Systems (ICCES)*. IEEE, 2019, pp. 28–33.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[33] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," *arXiv preprint arXiv:1908.03265*, 2019.

[34] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

A.4 Speech2Phone: A new and efficient method for training speaker recognition models

Título:	**Speech2Phone: A Novel and Efficient Method for Training Speaker Recognition Models**
Autores:	**Edresson Casanova, Arnaldo Candido Junior, Christopher Shulby, Frederico Santos de Oliveira, Lucas Rafael Stefanel Gris, Hamilton Pereira da Silva, Sandra Maria Aluisio, Moacir Antonelli Ponti**
Ano:	**2021**
Conferência:	**Brazilian Conference on Intelligent Systems (BRACIS)**
Situação:	**Publicado**

Contribuições relevantes do artigo:

1. Propõe um novo método para treinamento de modelos de verificação de locutores, promissor para cenários com poucos dados disponíveis;
2. O método proposto alcança desempenho similar aos modelos estado da arte utilizando 500 vezes menos dados.

Speech2Phone: A Novel and Efficient Method for Training Speaker Recognition Models

Edresson Casanova[1]*, Arnaldo Candido Junior[2], Christopher Shulby[3], Frederico Santos de Oliveira[4], Lucas Rafael Stefanel Gris[2], Hamilton Pereira da Silva[2], Sandra Aluisio[1], Moacir Antonelli Ponti[1]

[1] University of São Paulo, São Carlos, Brazil
[2] Federal University of Technology - Paraná, Medianeira, Brazil
[3] DefinedCrowd Corp., Seattle - WA, USA
[4] Federal University of Goias, Goiânia, Brazil

Abstract. In this paper we present an efficient method for training models for speaker recognition using small or under-resourced datasets. This method requires less data than other SOTA (State-Of-The-Art) methods, e.g. the Angular Prototypical and GE2E loss functions, while achieving similar results to those methods. This is done using the knowledge of the reconstruction of a phoneme in the speaker's voice. For this purpose, a new dataset was built, composed of 40 male speakers, who read sentences in Portuguese, totaling approximately 3h. We compare the three best architectures trained using our method to select the best one, which is the one with a shallow architecture. Then, we compared this model with the SOTA method for the speaker recognition task: the Fast ResNet–34 trained with approximately 2,000 hours, using the loss functions Angular Prototypical and GE2E. Three experiments were carried out with datasets in different languages. Among these three experiments, our model achieved the second best result in two experiments and the best result in one of them. This highlights the importance of our method, which proved to be a great competitor to SOTA speaker recognition models, with 500x less data and a simpler approach.

Keywords: speaker verification, speaker recognition, speaker identification

1 Introduction

Voice recognition is widely used in many applications, such as intelligent personal assistants [14], telephone-banking systems [4], automatic question response [12], among others. In several of these applications, it is useful to identify the speaker, as is the case in voice-enabled authentication and meeting loggers. Speaker verification can be done in two scenarios: open-set and closed-set. In both scenarios, the objective is to verify that two audio samples belong to the same speaker. However, in the closed-set scenario, the verification is restricted only to speakers seen during the training of the models. On the other hand, in the open-set scenario, verification occurs with speakers not seen in model training [11, 17]. The verification of speakers in an open-set scenario is especially desired in applications such as meeting loggers, since in these applications speakers can

* Corresponding author: edresson@usp.br

be added frequently, thus, the use of closed-set models would imply the retraining of the model after the insertion of new speakers.

The first works to use deep neural networks in speaker recognition in an open-set scenario learned speaker embeddings were [30, 29], using the Softmax function. Although the Softmax classifier can learn different embeddings for different speakers, it is not discriminatory enough [8]. To work around this problem, models trained with softmax were combined with back-ends built on Probabilistic Linear Discriminant Analysis (PLDA) [16] to generate scoring functions [26, 30]. On the other hand, Softmax Angular [19] was proposed and it uses cosine similarity as a logit entry for the softmax layer, and it proved to be superior to the use of softmax only.

Thereafter, Additive Margins in Softmax (AM-Softmax) [34] was proposed to increase inter-class variance by introducing a cosine margin penalty in the target logit. However, according to [8] training with AM-Softmax and AAM-Softmax [10] proved to be a challenge, as they are sensitive to scale and margin value in the loss function. The use of Contrastive Loss [7] and Triplet Loss [27, 5] also achieved promising results in speaker recognition, but these methods require careful choice of pairs or triplets, which costs time and can interfere with performance [8]. Finally, the use of Prototypical networks [35] for speaker recognition was proposed. Prototypical networks seek to learn a metric space in which the classification of open-sets of speakers can be performed by calculating distances to prototypical representations of each class. Generalized end-to-end loss (GE2E) [33] and Angular Prototypical (AngleProto) [8] follow the same principle and achieved state-of-the-art (SOTA) results recently in speaker recognition. Parallel with this work, [15] proposed the use of the AngleProto loss function in conjunction with Softmax, presenting a result superior to the use of AngleProto only. The authors proposed a new architecture presenting SOTA results.

In this work, we propose a new method for training speaker recognition models, called Speech2Phone. This method was trained with approximately 3.5 hours of speech and surpassed a model trained with 2.000 hours of speech using the GE2E loss function. Our method is based on the reconstruction of the pronunciation of a specific phoneme and has shown promise in scenarios with few available resources. In addition, the simplicity of its architecture makes our method suitable for real-time applications with low processing power.

Finally, to simplify the reproduction of this work, Python code and download links for the datasets used to reproduce all experiments are publicly available on the Github repository [5]

This work is organized as follows. Section 2 details the datasets used as well as the preprocessing performed to attend the proposed experiments. Section 3 presents the Speech2Phone method and experiments carried out to find the best model trained with this method for the identification of speakers in an open-set scenario. Section 4 compares the best model trained with the Speech2Phone method with the state of the art in the literature. Finally, Section 5 shows the conclusions of this paper.

[5] https://github.com/Edresson/Speech2Phone

2 Datasets and pre-processing

Section 2.1 presents the datasets used, as well as describes a new dataset created to attend the needs of our experiments. Sections 2.2 and 2.3 details the pre-processing performed on the datasets to allow the execution of all the proposed experiments.

2.1 Audio datasets

The VCTK [32] is an English language dataset with a total of 109 speakers. During its creation, the 109 speakers spoke approximately 400 sentences. The same phrases are spoken by all speakers. The dataset has approximately 44 hours of speech and is sampled at 48KHz.

Common Voice (CV) [2] is a massively multilingual transcribed speech dataset for research and development of speech technology. CV has 54 subsets and each of these sets have data from a language, currently the dataset has a total of 5,671 hours. In this work, we use version 5.2 of the corpus.

To train our method, it was necessary to build a specific dataset, which we call the Speech2Phone dataset. This dataset includes 40 male speakers, aged between 20 and 50 years. The dataset includes only Portuguese utterances, because that is the native language of the speakers. We chose to focus only on male speakers, because during the collection phase of our dataset we were able to collect only voices from 5 female speakers. To collect the data, each speaker was given a phonetically balanced text, according to the work of [28], which was comprised of 149 words. The reading time ranged from 42 to 95 seconds, totaling approximately 43 minutes of speech.

Additionally, we asked each speaker to say the phoneme /a/ for approximately three seconds. The central second of each capture was extracted and then used as expected output for the embedding models. The phoneme /a/ was chosen because it is simple to articulate and very frequent in the Portuguese language. The Speech2Phone dataset is publicly available on the Github repository[6]

2.2 Preprocessing of the Speech2Phone dataset

To preprocess the Speech2Phone dataset we extracted five-second speech segments from the original audio length. The five-second window was defined after preliminary experiments, varying the input duration. In order to maximize the number of speech segments, we used the overlapping technique, in which the window was shifted one second each time and an instance was extracted during the total audio duration. The main dataset resulted in 2,394 speech segments totaling 3 hours and 23 minutes of speech. The next step was to divide the dataset into smaller sets to attend the needs of each proposed experiment. Therefore, the original dataset was divided into four subsets. The Partitions A_1 and B_1 each have 20 different speakers and have approximately 1,097 samples each. Partitions A_2 and B_2 have approximately 100 speech segments each and have, respectively, speakers from the A_1 and B_1 partition. Thus, A partitions do not have speakers in common with B partitions.

[6] https://github.com/Edresson/Speech2Phone

2.3 Pre-processing of speaker verification datasets

We preprocessed the VCTK and CV datasets in order to use them for speaker verification. For the VCTK we chose to use the entire dataset and for CV we used the test subsets of the dataset in Portuguese (PT), Spanish (ES) and Chinese spoken in China (ZH).

The VCTK dataset has, in many of its samples, long initial and final silences. In order to ensure that this feature does not affect our analysis, we chose to remove these silences. So, we applied Voice Activity Detection (VAD) using the Python implementation of the Webrtcvad toolkit[7].

We used VCTK and CV to test our models. The datasets were used to build audio pairs. The positive class is composed from audio pairs from the same speaker, while negative class has pairs from different speakers. In this scenario, it is possible to build more examples from the negative class. To avoid class imbalance issues, we defined the maximum number of negative pairs analyzed as the number of positive samples divided by the number of speakers. We also removed speakers with less than two samples.

Table 1 shows the number of speakers, language and number of positive and negative speech segments of the datasets used to verify the speaker in our experiments. The Python code used for the preprocessing of the dataset, as well as the link to download the versions of the VCTK and CV datasets used are available in the Github repository[8].

Table 1: Preprocessed Speaker Verification datasets

Dataset	**Language**	**Nº Speakers**	**Nº Pos. Samples**	**Nº Neg. Samples**
Common Voice	PT	525	25,846	25,847
	ES	4,167	19,355	19,356
	ZH	1,968	14,656	14,657
VCTK	EN	109	9,084,638	9,001,368

3 Proposed Method: Speech2Phone

Section 3.1 presents the experiments carried out to choose the best model trained with the Speech2Phone method to identify speakers in the open-set scenario. On the other hand, section 3.2 presents the results of Speech2Phone experiments.

3.1 Speech2Phone Experiments

The goal of open-set models is to be speaker independent; additionally, a desirable feature is to be multilingual and text independent. In pursuit of these goals, we propose that the neural network training uses five second speech fragments as input and, as expected output, the reconstruction of a second of a simple phoneme (/a/ in our experiments). As

[7] https://github.com/wiseman/py-webrtcvad

[8] `https://github.com/Edresson/Speech2Phone/tree/master/Paper/EER-Experiments`

the phoneme sounds differ according to each speaker, a good reconstruction would allow the model to distinguish between speakers. Focusing on a single phoneme allows for dimensionality reduction in the embedding layer.

In the Speech2Phone experiments, the models input and expected output is represented in the form of Mel Frequency Cepstral Coeficients (MFCCs) [20]. We extract MFCCs using the Librosa [21] library. The default sampling rate (22KHz) was used. We chose to empirically extract 13 MFCCs. Windowed frames were used as defined by the default parameters in Librosa 0.6, namely, a 512 Hop Length and 2,048 as the window length for the Fast Fourier Transform [24]. In addition, as the models must reconstruct an MFCC segment they were induced using Mean Square Error (MSE) [13].

Several models using the Speech2Phone method, and consequently the Speech2Phone dataset, have been tested. We report the three most interesting ones in this section. To evaluate these experiments, we used the accuracy of the speaker identification.

To calculate the accuracy, we randomly chose and entered an embedding of a sample of each of the speakers in a database. For the calculation it is necessary to search the extracted embedding in an embeddings database. This is done by running the KNN [22] algorithm with $k = 1$ and using the Euclidean distance. If the embedding of a speaker has a Euclidean distance closer to its embedding registered in the database than the embedding of all other speakers, this will be counted as a hit, otherwise it will be counted as an error.

In the experiments, we compared the performance of the open-set models in a closed-set scenario. All of these experiments were trained with the Speech2Phone dataset A_1 subset, described previously in Section 2.2. Therefore, the models were trained with only 1,037 speech segments of 20 different speakers, approximately 1.5 hours of speech.

Tensorflow [1] and TFlearn [31] were used to generate the neural networks for all Speech2Phone Experiments. All models were trained using the Adam Optimizer [18]. The convolutional layers in all experiments have a stride of 1. Hyperparameters used to train each model are presented in Table 2.

Table 2: Experiments hyperparameters

Experiment	**Epochs**	**Learning Rate**	**Batch Size**
1 (Dense)	1,000	0.0007	128
2 (Conv)	10	0.0010	16
3 (Recurrent)	100	0.0050	256

To try to find the best topology for the Speech2Phone method, we propose the following experiments:

- **Experiment 1 (Dense)**: this experiment consists of a fully connected feed-forward neural network with one hidden layer for embedding extraction. The architecture of the model used in this experiment is shown in the Figure 1.
- **Experiment 2 (Conv)**: This experiment is based on Experiment 1, but it uses a fully convolutional neural network. This network has only convolutional layers, in addition the decoder uses upsampling layer. Following [8] we use 2D convolutions.

The speech segments can be seen as a bidimensional image matrix where columns are time steps and the rows are cepstral coefficients. Convolutions have the advantage of being translationally invariant in this matrix. As we try to reconstruct a specific phoneme, this is a desired property, since the location of the instance may contain the target phoneme may occur in different parts of the window. The model architecture used in this experiment is shown in the Figure 3.

- **Experiment 3 (Recurrent)**: This experiment is based on Experiment 1, but it consists of a recurrent fully-connected neural network for embedding generation. The five-second window is split into five segments containing one second each, in which the model analyzes one at a time. Considering that the recurrence window is small, we used classical recurrence instead of long term models like LSTM [13], since vanishing gradients are less prone to occur. If the phoneme of interest happens in one of the five fragments, the recurrence network should store it in its memory before reconstructing it in the final step, potentially improving the reconstruction accuracy. The approach reduces the number of learned parameters and, consequently, also improves training times. The model architecture used in this experiment is shown in the Figure 2.

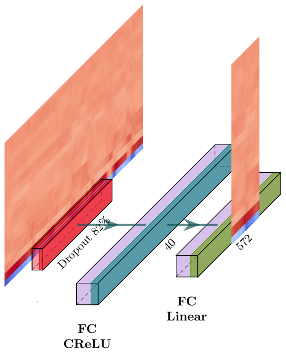


Fig. 1: Fully Connected Shallow Neural Network

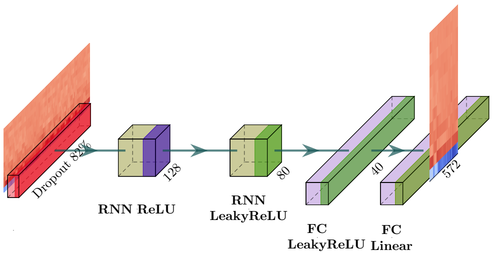


Fig. 2: Recurrent Neural Network

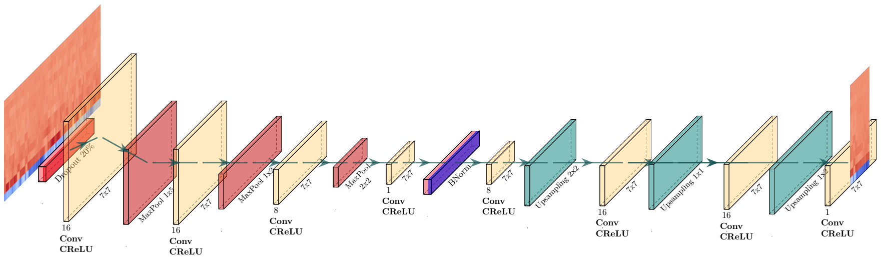


Fig. 3: Fully Convolutional Neural Network

3.2 Speech2Phone results

Table 3 shows the results of the Speech2Phone experiments, showing accuracy obtained in the open-set scenario, that is, evaluations using different speakers for training and testing the models and closed-set, where the training and test speakers are the same.

Table 3: Results of Speech2Phone Experiments

Experiment	Scenario (Subset)	Accuracy	Test speech segments
1 (Dense)	closed-set (A_2)	77.50	100
	open-set ($B_1 + B_2$)	76.96	1,197
2 (Conv)	closed-set (A_2)	100.00	100
	open-set($B_1 + B_2$)	64.43	1,197
3 (Recurrent)	closed-set (A_2)	88.75	100
	open-set ($B_1 + B_2$)	50.28	1,197

In this set of experiments, the neural models received 5 seconds of audio from a specific speaker and were induced to reconstruct 1 second of the phoneme /a/ in the voice of that speaker. The goal is to obtain good results with little training data in contrast to the GE2E and AngleProto loss functions, which need a large data set for good performance. To conduct the proposed experiments, samples from 20 speakers were used for training and samples from another 20 speakers for testing, as previously discussed in Section 3.1.

Experiment 1 explored the use of a fully connected neural network and in an open-set scenario, it obtained an accuracy of 76.96%, the best accuracy in the open-set scenario for all experiments. On the other hand, in the closed-set scenario, it obtained 77.50% accuracy. This was the worst result for all closed-set experiments. We believe that the fully connected model achieved the best results on the open-set due to the low number of parameters, thus being less prone to overfit. In addition, the dataset is very small and in this way, deeper models are very likely to memorize features dependent on the speaker or noise artifacts. Apparently, the recurrent and convolutional models specialized in extracting particular features for the speakers in the training set in order to reconstruct the output. In this way, as the deeper models learn specific characteristics of the speakers, their generalization for new speakers is impaired, having good performance in the closed-set scenario and a drop in performance in the open-set.

Experiment 2, which explored the use of a fully convolutional neural network for generating embeddings, presented the second best result (64.43%) in the open-set scenario. On the other hand, in the closed-set, the model achieved the best result obtaining an accuracy of 100%. Translational invariance is a useful feature from convolutional networks and can be used to detect specific phonemes independently of where their occur in the audio. We believe that convolutional models are suitable for this task; however, due to the low amount of data the models learn characteristics dependent on speakers, which leaves this model at a disadvantage in open-set scenario, having a worse performance than Experiment 1.

Experiment 3 explored the use of a recurrent neural network with fully connected layers for generating embeddings, resulting in the worst accuracy in the open-set scenario

(50.28%) and the second worst in the closed-set scenario (88.75%). Recurrent models can perform a more detailed analysis on the input audio, searching patterns in one input fragment at a time. However, a problem may happen when this pattern is split in a different analysis window. The simple recurrent model tested could not overcome this issue. We also evaluated, in preliminary experiments, a recurrent LSTM network, but it did not perform as well as simple recurrence, as there is no need for long-term memory in a 5-step analysis process.

In the open-set scenario, the superiority of the fully connected model (Experiment 1) is noticeable. This is because they are able to generalize better for new speakers and have proven to be less prone to overfitting for the task. In addition, the fully connected model was able to maintain very close accuracy in the closed-set scenario (77.50%), compared to the open-set scenario (76.96%), while the other models had a great drop in performance. On the other hand, the fully convolutional model (Experiment 2) also showed promising results with a performance 12.53% below the fully connected model.

4 Application: Speaker Verification

Section 4.1 presents the proposed experiments to compare our method with the state of the art in the literature. On the other hand, section 3.2 presents the results of speaker verification experiments.

4.1 Speaker verification Experiments

An important question is how well a model trained with the Speech2Phone method performs in speaker verification. To answer this question we compared Speech2Phone with the Fast ResNet–34 model proposed by [8] trained using the Angular Prototypical [8] and GE2E [33] losses function. We chose this model because it presents state-of-the-art results in the VoxCeleb [23] dataset, in addition the authors made the pre-trained models available in the Github repository[9].

To compare the models, we chose two datasets, one of which is multi-language, presented in Section 2.3. As Speech2Phone receives 22 kHz sample rate audio, while Fast ResNet–34 16 kHz audio input. We cannot directly compare the results of the models in the VoxCeleb test dataset. As voxceleb is sampled at 16 kHz, it would not be a fair comparison to resample audios from 16 kHz to 22 kHz. Therefore, we chose other datasets with a sample rate greater than 22 kHz. In addition, using the dataset test that the Fast ResNet–34 model was trained on could put the Speech2Phone model at a disadvantage.

The audios for each dataset were resampled to 16 kHz for the test with the Fast ResNet–34 model and to 22 kHz for the test with Speech2Phone, thus making a fair comparison between the models. For comparison, we use the metric Equal Error Rate (EER) [6], the lower the EER is, the greater the accuracy becomes.

To compare the best model trained with the Speech2Phone method with the SOTA we propose the following experiments:

[9] https://github.com/clovaai/voxceleb_trainer

- **Experiment 4 (Speech2Phone)**: This experiment uses the same model and hyperparameters as experiment 1; however, the model was trained with the entire Speech2Phone dataset. Therefore, the model was trained with 2,394 speech segments of 40 speakers totaling approximately 3 hours and 23 minutes of speech.
- **Experiment 5 (Fast AngleProto)**: This experiment uses the Fast ResNet–34 model, proposed in [8], this model is trained with the Angular Prototypical loss function in the Voxceleb dataset, which has approximately 2,000 hours of speech by approximately 7,000 speakers. This model achieves an EER of approximately 2.2 in the voxceleb dataset test set, as reported in [8].
- **Experiment 6 (Fast GE2E)**: This experiment also uses the Fast ResNet–34 model, and was also trained with the same number of speakers and hours of experiment 5. However the model is trained using the GE2E loss function.

Unlike the Fast ResNet–34 model, which accepts audios of varying sizes, our model receives an MFCC of just 5 seconds of speech as input. However, in the VCTK and CV datasets we have audios with variable sizes. Therefore, for a fair comparison and each model to have access to all the audio content for calculating the EER of the Speech2Phone model, we proceed as follows. For audios longer than 5 seconds, we use the overlapping technique, as described in Section 2.2. Therefore, after applying the overlapping technique for a six-second audio, two five-second samples are obtained, the resulting embedding for that sample is the average between the predicted embedding for these two five-second samples. On the other hand, for sample less than 5 seconds, we repeat the audio frames until reaching at least 5 seconds, for example, a three-second audio is repeated once, thus obtaining a six-second audio, where the resulting audio applies the overlapping technique, since we audios longer than 5 seconds.

4.2 Speaker Verification Results

In this set of experiments we compared the performance of our best experiment trained with the Speech2Phone method with the Fast ResNet–34 model proposed by [8], trained with approximately 2,000 hours of speech using the Angular Prototypical loss function [8] and GE2E [33]. Table 4 shows the EER of these experiments in English using the VCTK dataset, in Portuguese (PT), Spanish (ES) and Chinese (ZH) using the Common Voice dataset.

Experiment 4, which consisted of the best Speech2Phone experiments trained using the entire Speech2Phone dataset, that is, approximately 3 hours and 23 minutes of speech from 40 different speakers. This experiment achieved the best EER of all experiments in the VCTK dataset. For the Common Voice dataset, the model achieved the second best result for the PT and ZH subsets, being surpassed only by the Fast ResNet–34 model trained with the Angular Prototypical loss function (Experiment 5). On the other hand, in the ES subset, this experiment had the worst performance of all experiments.

Experiment 5, which consisted of the Fast ResNet–34 model trained with the Angular Prototypical loss function, obtained the second best EER in the VCTK dataset, only surpassed by experiment 4 which was trained with the Speech2Phone method. In addition, this experiment achieved the best result in all subsets of the Common Voice dataset.

Experiment 6, which consisted of the Fast ResNet–34 model trained with the GE2E loss function, obtained the worst EER in the VCTK dataset. This experiment also had the

Table 4: Results for speaker verification experiments

Experiment	**Datasets**	**EER (%)**
4 (Speech2Phone)	VCTK	**22.7041**
	CV PT	13.6805
	CV ZH	10.3909
	CV ES	7.7551
5 (Fast AngleProto)	VCTK	23.8011
	CV PT	**7.2468**
	CV ZH	**7.2666**
	CV ES	**2.8622**
6 (Fast GE2E)	VCTK	27.0647
	CV PT	14.0751
	CV ZH	12.9563
	CV ES	5.0530

worst EER in the ZH and PT subsets of Common Voice. However, the model achieved the second best EER in the ES subset of common voice.

Experiment 4 showed the potential of the Speech2Phone approach, which despite having been trained with only 3 hours and 23 minutes of speech from only 40 speakers, managed to better results in 3 of the 4 evaluated subsets. Experiment 6, which is the Fast ResNet–34 model trained with 2000 hours of speech and approximately 7,000 speakers using the GE2E loss function. In addition, this experiment was able to perform better than both Experiments 5 and 6 in the VCTK dataset.

Experiment 6 compared to Experiment 4, achieved a better result in 3 of the 4 evaluated subsets, this was already expected due to the difference of more than 1.996 hours of speech and 6.960 speakers between the training datasets of the two models. In addition, the Experiment 4's training dataset has only male voices, as the reconstruction of a female voice is different and the VCTK and Common voice datasets have female speakers this should cause a decrease of performance at test time. Another point is that the training dataset used in Experiment 4 is a high quality dataset that has a low amount of background noise, so the model probably did not learn to ignore noise and the reconstruction of the /a/ phoneme is harmed. Despite this quality and absence of noise facilitating learning during training, the model is impaired in a noisy situation. Another indication of this problem can be seen by its better performance than all the other experiments in the VCTK dataset, which is a high quality and low noise, built for speech synthesis and voice transfer applications. The performance of the model drops, compared to experiment 5, when the model is used in audios recorded in uncontrolled environments such as Commom Voice.

In Experiments 5 and 6 we verified what has already been shown in [8], that the Fast ResNet–34 model trained with the AngleProto loss function in the Voxceleb dataset obtains an EER higher than the same model trained with the GE2E loss function. However, the authors in their work compared the models only in the VoxCeleb test dataset, we on the other hand, made a comparison using 4 different languages and different datasets, thus making a broader comparison.

An important consideration is that given the way we propose our experiments, we cannot say that language is a factor that decreases or increases the performance of the models. The high values of EER in the English language, for example, are due to the way the speaker verification dataset was set up. The VCTK has only 109 speakers and many samples were considered for each speaker as can be seen in Table 1. A greater number of test instances make the task more difficult and tend to increase the EER values. Therefore, we can only compare the individual performance of each model in the datasets and we cannot discuss decrease or increase in performance with the language change.

5 Conclusions and future work

In this article, we proposed a novel training method for speaker recognition models, called Speech2Phone. To enable the training of this method, we also built a novel dataset. Fully connected, fully convolutional and recurring models were explored. We observed that the fully connected models have a better performance in open-set scenarios, although they have had the worst performance for the closed-set scenario, while convolutional models have a better performance in the closed-set scenario, but they do not generalize well for the open-set scenario. The best model in our experiments was trained on 3 hours and 23 minutes of speech and compared to two SOTA models in the VoxCeleb dataset, which were trained with approximately 2000 hours of speech. The results obtained were comparable to SOTA even with an amount of data approximately 500 times smaller.

This work contributes directly to the area of speaker recognition, presenting a promising method for training speaker recognition models. In addition, the model proposed here can be used in tasks such as speech synthesis [25], voice cloning [3] and multilingual speech conversion [36]. In these tasks, the speaker recognition system embeddings are used to represent the speaker. In addition, an advantage of Speech2Phone in relation to the models proposed in [8] is the speed of execution, as our best model is a fully connected shallow neural network, making it faster. This feature is very desirable for applications due to the need to run in real time.

As our model demands a specific dataset format, we were not able to evaluate its training in a large dataset. We plan to address this issue in future works. For this we intend to increase the model's training dataset as much as possible, and make it public. Additionally, we intend to explore the use of multispeaker speech synthesis [9] and voice cloning [3] to generate a dataset with more speakers and a larger vocabulary. On the other hand, we intend to investigate the possibility of a hybrid method that uses a Speech2Phone technique and Angular Prototypical loss function, thus being able to learn from a larger amount of data and at the same time guide the model's learning with the reconstruction of a phoneme. In addition, we intend to explore the insertion of noise in the training dataset in order to make the model more robust to noise.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)

2. Ardila, R., Branson, M., Davis, K., Kohler, M., Meyer, J., Henretty, M., Morais, R., Saunders, L., Tyers, F., Weber, G.: Common voice: A massively-multilingual speech corpus. In: Proceedings of The 12th Language Resources and Evaluation Conference. pp. 4218–4222 (2020)
3. Arik, S., Chen, J., Peng, K., Ping, W., Zhou, Y.: Neural voice cloning with a few samples. In: Advances in Neural Information Processing Systems. pp. 10019–10029 (2018)
4. Bowater, R.J., Porter, L.L.: Voice recognition of telephone conversations (Aug 21 2001), US Patent 6,278,772
5. Bredin, H.: Tristounet: triplet loss for speaker turn embedding. In: Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. pp. 5430–5434. IEEE (2017)
6. Cheng, J.M., Wang, H.C.: A method of estimating the equal error rate for automatic speaker verification. In: 2004 International Symposium on Chinese Spoken Language Processing. pp. 285–288. IEEE (2004)
7. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). vol. 1, pp. 539–546. IEEE (2005)
8. Chung, J.S., Huh, J., Mun, S., Lee, M., Heo, H.S., Choe, S., Ham, C., Jung, S., Lee, B.J., Han, I.: In defence of metric learning for speaker recognition. Proc. Interspeech 2020 pp. 2977–2981 (2020)
9. Cooper, E., Lai, C.I., Yasuda, Y., Fang, F., Wang, X., Chen, N., Yamagishi, J.: Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6184–6188. IEEE (2020)
10. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4690–4699 (2019)
11. Ertaş, F.: Fundamentals of speaker recognition. Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi **6**(2-3) (2011)
12. Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A.A., Lally, A., Murdock, J.W., Nyberg, E., Prager, J., et al.: Building watson: An overview of the deepqa project. AI magazine **31**(3), 59–79 (2010)
13. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016), http://www.deeplearningbook.org
14. Gruber, T.R.: Siri, a virtual personal assistant—bringing intelligence to the interface (2009)
15. Heo, H.S., Lee, B.J., Huh, J., Chung, J.S.: Clova baseline system for the voxceleb speaker recognition challenge 2020. arXiv preprint arXiv:2009.14153 (2020)
16. Ioffe, S.: Probabilistic linear discriminant analysis. In: European Conference on Computer Vision. pp. 531–542. Springer (2006)
17. Kekre, H., Kulkarni, V.: Closed set and open set speaker identification using amplitude distribution of different transforms. In: 2013 International Conference on Advances in Technology and Engineering (ICATE). pp. 1–8. IEEE (2013)
18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
19. Liu, W., Wen, Y., Yu, Z., Li, M., Raj, B., Song, L.: Sphereface: Deep hypersphere embedding for face recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 212–220 (2017)
20. Logan, B., et al.: Mel frequency cepstral coefficients for music modeling. In: Ismir. vol. 270, pp. 1–11 (2000)

21. McFee, B., Raffel, C., Liang, D., Ellis, D.P., McVicar, M., Battenberg, E., Nieto, O.: librosa: Audio and music signal analysis in python. In: Proceedings of the 14th python in science conference. pp. 18–25 (2015)
22. Mitchell, R., Michalski, J., Carbonell, T.: An artificial intelligence approach. Springer (2013)
23. Nagrani, A., Chung, J.S., Zisserman, A.: Voxceleb: A large-scale speaker identification dataset. Proc. Interspeech 2017 pp. 2616–2620 (2017)
24. Nussbaumer, H.J.: The fast fourier transform. In: Fast Fourier Transform and Convolution Algorithms, pp. 80–111. Springer (1981)
25. Ping, W., Peng, K., Gibiansky, A., Arik, S.O., Kannan, A., Narang, S., Raiman, J., Miller, J.: Deep voice 3: Scaling text-to-speech with convolutional sequence learning. In: International Conference on Learning Representations (2018)
26. Ramoji, S., Krishnan V, P., Singh, P., Ganapathy, S.: Pairwise discriminative neural plda for speaker verification. arXiv preprint arXiv:2001.07034 (2020)
27. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)
28. Seara, I.: Estudo Estatístico dos Fonemas do Português Brasileiro Falado na Capital de Santa Catarina para Elaboração de Frases Foneticamente Balanceadas. Ph.D. thesis, Dissertação de Mestrado, Universidade Federal de Santa Catarina … (1994)
29. Snyder, D., Garcia-Romero, D., Povey, D., Khudanpur, S.: Deep neural network embeddings for text-independent speaker verification. In: Interspeech. pp. 999–1003 (2017)
30. Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S.: X-vectors: Robust dnn embeddings for speaker recognition. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5329–5333. IEEE (2018)
31. Tang, Y.: Tf. learn: Tensorflow's high-level module for distributed machine learning. arXiv preprint arXiv:1612.04251 (2016)
32. Veaux, C., Yamagishi, J., MacDonald, K., et al.: Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit. University of Edinburgh. The Centre for Speech Technology Research (CSTR) (2016)
33. Wan, L., Wang, Q., Papir, A., Moreno, I.L.: Generalized end-to-end loss for speaker verification. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4879–4883. IEEE (2018)
34. Wang, F., Cheng, J., Liu, W., Liu, H.: Additive margin softmax for face verification. IEEE Signal Processing Letters **25**(7), 926–930 (2018)
35. Wang, J., Wang, K.C., Law, M.T., Rudzicz, F., Brudno, M.: Centroid-based deep metric learning for speaker recognition. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3652–3656. IEEE (2019)
36. Zhou, Y., Tian, X., Xu, H., Das, R.K., Li, H.: Cross-lingual voice conversion with bilingual phonetic posteriorgram and average modeling. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6790–6794. IEEE (2019)

ICMC USP

SÃO CARLOS