



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO DE CIÊNCIAS DA SAÚDE
DEPARTAMENTO DE FONOAUDIOLOGIA
CURSO DE FONOAUDIOLOGIA

Kauana Knappmann

**Processamento de arquivos de áudio para composição de testes de
reconhecimento de fala em português**

Florianópolis

2024

Kauana Knappmann

**Processamento de arquivos de áudio para composição de testes de
reconhecimento de fala em português**

Trabalho de Conclusão de Curso submetido ao curso de Fonoaudiologia do Centro de Ciências da Saúde da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Bacharela em Fonoaudiologia.

Orientador(a): Prof.Dr. Stephan Paul

Florianópolis

2024

Knappmann, Kauana

Processamento de arquivos de áudio para composição de testes de reconhecimento de fala em português / Kauana Knappmann ; orientador, Stephan Paul, 2024.

22 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro de Ciências da Saúde, Graduação em Fonoaudiologia, Florianópolis, 2024.

Inclui referências.

1. Fonoaudiologia. 2. Audiologia. 3. Processamento de Sinais Assistido por Computador. 4. Testes Auditivos. I. Paul, Stephan. II. Universidade Federal de Santa Catarina. Graduação em Fonoaudiologia. III. Título.

RESUMO

Introdução: Testes de reconhecimento de fala são instrumentos relevantes para a avaliação audiológica de portadores de perda auditiva. A criação de testes de reconhecimento de fala em português é incipiente, parcialmente devido à escassez de recursos para tal. **Objetivo:** Desenvolver códigos computacionais para modificação automatizada de grandes quantidades de gravações de sentenças, que possam realizar modificações de formato, filtragens, simular o processamento dos sinais sonoros em implantes cocleares e ajustar as médias quadráticas de amplitude para equalizar o volume sonoro percebido entre sentenças. **Metodologia:** Para os diferentes processamentos pretendidos foram desenvolvidos códigos em Python, usando as interfaces Spyder e pacotes tais como o pydub, soundfile, os e numpy. Os códigos foram testados em dois conjuntos de arquivos de áudio gravados previamente em português brasileiros, no formato de .MP3 e .WAV. **Resultados:** Foram implementados códigos para 1) modificação do formato do arquivo, 2) ajuste do *fade-in* e *fade-out*, 3) filtragem de passa-alta, 4) vocoderização opcional e 5) ajuste das médias quadráticas das amplitudes. Os testes dos códigos desenvolvidos em dois conjuntos de sentenças disponíveis em .WAV e .MP3 na língua portuguesa demonstraram resultados consistentes com o esperado. **Conclusão:** Desenvolveu-se códigos na linguagem Python para modificação de maneira automatizada de arquivos de áudio, disponíveis no site GitHub para adaptações e aprimoramentos por terceiros.

DESCRITORES

Audiologia, Processamento de Sinais Assistido por Computador, Testes Auditivos.

ABSTRACT

Introduction: Speech recognition tests are relevant tools for the audiological evaluation of individuals with hearing loss. The development of speech recognition tests in Portuguese is still in its early stages, partly due to the lack of resources for such endeavors. **Objective:** To develop computational codes for the automated modification of large quantities of sentence recordings, capable of performing format modifications, filtering, simulating sound signal processing in cochlear

implants, and adjusting root mean square amplitude to equalize perceived volume between sentences. **Methodology:** Python codes were developed for the intended processes, using the Spyder interface and packages such as pydub, soundfile, os, and numpy. The codes were tested on two sets of previously recorded audio files in Brazilian Portuguese, in .MP3 and .WAV formats. **Results:** Codes were implemented for 1) file format modification, 2) fade-in and fade-out adjustment, 3) high-pass filtering, 4) optional vocoderization, and 5) adjustment of root mean square amplitude. Testing the developed codes on two sets of sentence recordings available in .WAV and .MP3 formats in Portuguese showed consistent results as expected. **Conclusion:** Python codes were developed for the automated modification of audio files, available on the GitHub website for further adaptations and improvements by third parties.

KEYWORDS

Audiology, Signal Processing, Computer-Assisted, Hearing Tests.

1. INTRODUÇÃO

A hipoacusia, também chamada de perda auditiva, é uma deficiência do sistema sensorial auditivo que afeta a qualidade de vida e a comunicação¹. De acordo com a Organização Mundial da Saúde (OMS)², há uma estimativa de que até o ano de 2050 aproximadamente 2,5 bilhões de pessoas ao redor do mundo terão algum tipo de hipoacusia, e que cerca de 700 milhões vão necessitar de algum tipo de tratamento, a menos que medidas sejam tomadas³.

Para avaliar a habilidade de compreensão e interpretação de fala em diferentes condições e para diferentes grupos de sujeitos, por exemplo com e sem perda auditiva, existem diferentes testes disponíveis, chamados de testes de reconhecimento de fala⁴. Eles podem ser bastante eficientes para avaliação de candidatos ou usuários de implantes cocleares⁵. Variações deste tipo de teste podem incluir simulações da degradação auditiva ou ruídos competitivos, para reproduzir obstáculos auditivos que os usuários podem encontrar em locais ruidosos e em ambientes onde a qualidade do som é comprometida⁶.

Ao buscar por avaliações de reconhecimento de fala padronizadas para o Português Brasileiro, evidenciou-se uma escassez de opções, o que pode influenciar na qualidade dos resultados obtidos. De acordo com um estudo⁷, os testes utilizados no Brasil são o HINT adaptado para a língua portuguesa, também há o nomeado de Lista de Sentenças em Português e ainda o nomeado de Listas de Sentenças do Centro de Pesquisas Audiológicas (CPA). Estes poucos recursos padronizados disponíveis representam um desafio para profissionais que procuram ferramentas validadas especificamente para a língua portuguesa⁸.

A falta de mais testes é causada por vários fatores, entre eles os desafios envolvidos na criação destes testes, ao começar pela necessidade de criar centenas de sentenças foneticamente balanceadas, gravar todas as sentenças com diferentes locutores e processar todas as sentenças para que possam compor um conjunto de sentenças organizadas em listas com características bem definidas, tal como volume equalizado e dificuldade de reconhecimento padronizado.

Para a criação de testes de reconhecimento de fala, são necessários procedimentos que analisam e modificam grandes números de arquivos de áudio

previamente gravados, de acordo com regras específicas para atingir determinados objetivos, como por exemplo gerar arquivos modificados por processamento de sinais⁹ tais como filtragem de frequências indesejadas, equalização de volume, eliminação ou inserção de pausas, suavização de transições, entre outros.

No que concerne ao desenvolvimento de testes de reconhecimento de fala para usuários de implante coclear, podem ser desejáveis ainda modificações nas sentenças que simulam o processamento pelo implante coclear ou pela interface eletrodo-nervo auditivo. O Vocoder é um método de processamento de sinais que simula por exemplo o processamento de sinal em um implante coclear, replicando a transformação realizada no som para torná-lo perceptível para pessoas com deficiência auditiva neurosensorial¹⁰.

Tendo em vista a complexidade de processamento necessário em grandes números de sentenças, especialmente durante o processo de desenvolvimento de testes, fica evidente que etapas de processamento precisam ser automatizadas, especialmente tendo em vista testes de reconhecimento de fala voltados para usuários de implante coclear, já que para o desenvolvimento destes é necessário o processamento pelo Vocoder, além das demais etapas previamente descritas.

Assim, justifica-se o número pequeno de testes de reconhecimento de fala voltados para esta aplicação específica, sendo que o teste AzBio é o mais utilizado usado para a avaliação da compreensão de fala de deficientes auditivos, e particularmente usuários de implantes cocleares¹¹.

O teste de sentenças de reconhecimento de fala nomeado de Arizona Biomedical Institute Sentences Test - AzBio, foi inicialmente desenvolvido para deficientes auditivos e usuários de implante coclear falantes da língua inglesa, publicado pela primeira vez em 2004 e validado por Spahr e Dorman, pesquisadores do Departamento de Speech and Hearing Science da Arizona State University nos Estados Unidos da América em 2012 ^{12 13}.

Por já ser usado em diversos países de línguas diferentes, mostra-se uma boa opção para o português do Brasil, podendo oferecer avaliação precisa e confiável, preenchendo uma lacuna que existe nos testes validados. Com isso pode contribuir significativamente nas pesquisas e avanços na área trazendo melhorias nos diagnósticos e avaliações relacionadas com a audição e a fala.

Para viabilizar o desenvolvimento do teste AzBio em Português do Brasil o presente trabalho implementou rotinas computacionais de processamento de sinais, que realizam de forma automatizada as mais diferentes tarefas necessárias para manipulação das sentenças gravadas para compor a futura versão brasileira do AzBio.

2. MÉTODO

O material de áudio gravado em português brasileiro que está disponível para o presente trabalho consiste em dois conjuntos grandes de sentenças, gravadas por terceiros. O conjunto A de sentenças compreende 987 sentenças gravadas em um estúdio profissional. Para a gravação, as sentenças foram pronunciadas por 4 falantes, sendo dois homens e duas mulheres. O primeiro falante masculino gravou 248 sentenças, enquanto o segundo gravou 231. Da parte das falantes femininas, a primeira registrou 257 sentenças e a segunda 251. As gravações foram disponibilizadas para o estudo objeto deste artigo em formato .MP3, com taxa de amostragem em 44100 Hz.

O conjunto B consiste em 106 gravações de sentenças para a construção de uma avaliação da percepção de fala para crianças falantes do português brasileiro, pronunciadas por somente uma falante feminina. Estas gravações foram para o estudo objeto deste artigo em formato .WAV com taxa de amostragem de 44100 Hz.

Os dois conjuntos representam conjuntos de sentenças recentes, que estão sendo consideradas para integrar testes de reconhecimento de fala em desenvolvimento.

Tendo em vista que um grande número de sentenças, gravadas em arquivos de áudio, precisa ser manipulado em vários aspectos, todos os procedimentos necessários foram implementados na linguagem computacional Python, a fim de permitir um processamento automatizado deste grande número de arquivos.

Para desenvolver os códigos em Python destinados à modificação dos arquivos de sentenças, utilizou-se o ambiente de desenvolvimento de software Spyder, que é integrante do pacote de distribuição Anaconda. O Spyder oferece

uma interface (Figura 1) de desenvolvimento interativa e com funcionalidades de depuração que facilitaram a criação de códigos.

Como é comum na linguagem de programação Python, foram utilizadas várias bibliotecas e pacotes para realização de funções específicas de forma robusta e versátil. Estas bibliotecas são coleções de módulos que possuem várias funções, métodos já pré-definidos, além de classes, para executar tarefas específicas de forma eficiente. Elas fazem com que possam ser utilizados códigos já existentes, simplificando processos ¹⁴.

A título de exemplo, as bibliotecas *soundfile* e *pydub* foram usadas para uma manipulação apropriada dos arquivos de áudio, enquanto as bibliotecas *os* e *numpy* serviram para a gestão e a manipulação dos caminhos dos arquivos e *arrays* numéricos respectivamente. Já a biblioteca *scipy* foi usada para funcionalidades estatísticas, auxiliando a análise e o processamento dos dados sonoros. Todas estas bibliotecas estão disponíveis online ¹⁵.

3. RESULTADOS

3.1. ETAPAS DE PROCESSAMENTO DOS MATERIAIS DE ÁUDIO

Visando o processamento dos arquivos de áudio já existentes para integrarem futuros testes de reconhecimento de fala voltados para normo ouvintes, deficientes auditivos ou para usuários de implante coclear, todos os processamentos necessários foram organizados em diferentes etapas, conforme mostra o esquemático da Figura 2. Algumas das etapas são executadas de forma obrigatória, enquanto que outras etapas são facultativas, como por exemplo a vocoderização, que só é precisa caso se queira simular o efeito de um implante coclear.

Arquivos de áudio que inicialmente encontram-se em formato de áudio .MP3 devem passar por uma conversão para o formato .WAV, funcionalidade implementada através de códigos programáveis. Isso deve ocorrer porque devido ao formato .WAV ser *loss/less*, que não compromete a qualidade do áudio durante o armazenamento ou processamento, diferentemente do .MP3, que comprime o arquivo, removendo informações acústicas para reduzir o tamanho do arquivo. No entanto, é importante ressaltar que uma vez que o áudio foi comprimido no

formato .MP3, as informações removidas durante a compressão não podem ser recuperadas ao converter o arquivo para .WAV ¹⁶.

Na etapa seguinte, já com arquivos em formato .WAV, ocorre a suavização da transição entre o silêncio e o início da sentença (*fade-in*) e o final da sentença e o silêncio (*fade-out*), fazendo ainda exclusão da parte em que há silêncio excessivo no início e no fim do sinal. Essas ações são importantes para diminuir interferências, uma vez que silêncios prolongados podem ser contaminados por ruídos de baixa frequência, como o de 60 Hz, originado da rede elétrica.

O *fade-in* com duração de 1 milissegundo aumenta gradualmente as amplitudes do começo da sentença, fazendo com que haja uma transição suave entre os sons, enquanto o *fade-out* também de 1 milissegundo faz a mesma coisa no fim da sentença. Este processo traz uma experiência auditiva mais natural e agradável, diminuindo mudanças abruptas de áudio.

Para dar continuidade ao processamento dos arquivos, uma etapa de filtragem por meio de um filtro passa-alta foi implementada. A filtragem permite a passagem das frequências mais altas, atenuando as frequências mais baixas. A Figura 3 exemplifica o processo.

O filtro implementado é do tipo Butterworth de ordem 5, usado para deixar a resposta da amplitude do sinal acima da frequência de corte o mais plana possível e ao mesmo tempo tendo boa qualidade na transição da resposta em frequência (FRF) entre a banda de rejeição e a de passagem, conforme se verifica na Figura 3. A frequência de corte do filtro foi definida como 80Hz, ou seja frequências abaixo de 80Hz serão substancialmente atenuadas, e frequências levemente superiores a 80 Hz serão pouco afetadas, já que a FRF em 80 Hz passará pelo ponto de - 3 dB.

Esse ponto de corte foi escolhido de forma estratégica para não impactar de forma negativa as frequências fundamentais da voz humana, que são de aproximadamente 110 Hz para homens e de cerca de 220 Hz para mulheres. Essa escolha permite que o filtro minimize a interferência causada por ruídos de baixa frequência, sem afetar a inteligibilidade e naturalidade das vozes.

A próxima etapa, que os códigos podem executar, envolve a vocoderização das sentenças com objetivo de simular o processamento dentro de um implante coclear, criando arquivos sonoros que simulem a reprodução auditiva que pareça com a experiência sensorial desses dispositivos.

O processo denominado de vocoderização das sentenças, em que ocorre o processamento dos arquivos por um vocoder, é realizado por meio do software AngelSim, um software projetado para simular o processamento de sinal de um implante coclear.

As configurações para aplicação de vocoder em sentenças pode variar de acordo com o objetivo da pesquisa, e são portanto configuráveis. Para fins de exemplo, para a construção deste artigo, o vocoder foi parametrizado como processador de ruído/ onda senoidal com onda senoidal como onda portadora. O número de canais, que pode ser ajustado entre 1 e 32, foi definido como 5 para este projeto.

Ainda, podem ser parametrizados o mapeamento (a distribuição das frequências) ao longo do arranjo de eletrodos, por exemplo pelo mapeamento de Greenwood, utilizado por este estudo, a faixa de frequência desejada para o filtro portador pode ser configurado e neste caso utilizamos os parâmetros de 200 Hz a 7000 Hz e 24 dB/oitava, entre outros detalhes.

O próximo passo é a inserção de um intervalo de tempo antes de iniciar a sentença e também ao fim da mesma. Esta pausa serve para que o ouvinte possa se preparar para a próxima sentença reproduzida em situações nas quais várias sentenças sejam reproduzidas sequencialmente.

O último passo antes de ter as sentenças modificadas completamente, compreende o ajuste do Root Mean Square (RMS) da variável dependente do arquivo de áudio, apresentado de maneira figurada pela Figura 4. O RMS é uma medida estatística da amplitude média quadrática do sinal, e a modificação desta, afeta a percepção do volume sonoro.

Por meio do ajuste da amplitude média quadrática dos sinais de áudio é possível proporcionar que a sensação de volume sonoro produzida por cada sinal seja a mesma, o que é relevante em testes de reconhecimento de fala, já que o percentual de reconhecimento depende da razão sinal ruído, e portanto da potência do sinal de áudio, apresentado.

Como de costume para sinais de áudio, a amplitude média quadrática do sinal é definida em termos de uma escala logarítmica relativa dBFS (*dB full scale*). Cabe notar que o nível de pressão sonora de reprodução, em termos de dB SPL, dependerá do software e dos equipamentos utilizados para reprodução. Entretanto, assumindo linearidade e invariância no tempo deste sistema de

reprodução, o fato de ter arquivos sonoros com a mesma amplitude média quadrática expressa na escala dBFS garante que na reprodução o nível de pressão sonora de reprodução dB SPL também seja o mesmo.

3.2. VALIDAÇÃO DOS CÓDIGOS

Os códigos desenvolvidos e descritos na Seção 3.1. ETAPAS DE PROCESSAMENTO DOS MATERIAIS DE ÁUDIO realizaram modificações nos arquivos de áudio que podem ser observados não somente auditivamente, mas também por meio de representações das características dos arquivos de áudio antes de depois de cada modificação.

No que concerne a aplicação do *fade-in* e do *fade-out*, a Figura 5 representa o histórico no tempo de um arquivo exemplo antes e depois do processamento, marcando a diferença entre a cor cinza para a sentença antes de qualquer modificação e a cor azul representa a sentença com *fade-in* e *fade-out*. As diferenças observadas são bastante relevantes, evidenciando que o *fade-in* e o *fade-out* atuam sobre as amplitudes no início e do fim do sinal de áudio.

O processo de filtragem por meio do filtro passa-alta pode ser evidenciado da melhor forma por meio da comparação dos espectrogramas do sinal de áudio antes (Figura 6) e depois (Figura 7) da aplicação do filtro. Verifica-se pelo espectrograma da Figura 6 que o arquivo exemplo original está contaminado com energia de frequências baixas (<80Hz), e que esta energia de baixa frequência está inclusive presente antes e depois do sinal da fala. A parte que contém a fala está indicada por uma caixa vermelha. Já o espectrograma da Figura 7 mostra que o filtro removeu de forma satisfatória a contaminação em baixa frequência, preservando as frequências do sinal de fala.

A aplicação, opcional, do vocoder aos arquivos de áudio modifica os arquivos de áudio para simular as condições auditivas de um usuário de implante coclear. A vocoderização cria modificações consideráveis no espectro e na definição temporal dos sons, aproximando-se com a maneira que um implante coclear apresenta os sons ao usuário de um implante coclear. As Figuras 8 e 9 apresentam, por meio do histórico no tempo, um sinal exemplo e a modificação do mesmo após o processamento do vocoder no arquivo. A Figura 10 apresenta o espectrograma do sinal de áudio depois da passagem pelo vocoder, com as

modificações feitas, evidenciando as mudanças no sinal em comparação ao espectrograma apresentado na Figura 7, principalmente a perda de resolução em frequência típica em sinais processados por implantes cocleares.

O ajuste do valor médio quadrático (RMS) das amplitudes do sinal procura garantir que a sensação de volume sonoro seja a mesma para todos os sinais de áudio, ainda que em aproximação.

Conforme se verifica na Figura 4 o processo de ajuste dos valores RMS compreende várias etapas: carregamento dos arquivos, conversão os dados do sinal de integer para ponto flutuante, cálculo do RMS atual, definição do valor de RMS desejado, cálculo do fator de escala para ajustar o RMS ao valor desejado e aplicação da correção às amplitudes do sinal. Também é adicionado um intervalo de tempo em que as amplitudes são nulas no início e no final do áudio processado, para depois, ter o arquivo salvo. A adição deste intervalo de silêncio ocorre após o ajuste dos valores RMS, para não influenciar nos valores RMS ajustados.

Para avaliar o funcionamento correto do código que ajusta da média quadrática das amplitudes (ajuste RMS) para nível alvo de -20 dBFS, foram processadas os 2 conjuntos de sentenças do português brasileiro disponíveis para teste dos códigos. O conjunto A foi processado tanto sem passar por vocoderização como também após passagem pelo processo de vocoderização. Já o conjunto B foi processado apenas sem vocoderização.

A Figura 11 representa a distribuição das médias quadráticas das amplitudes do conjunto de 987 arquivos do conjunto A de sentenças antes do ajuste do RMS. Verifica-se uma grande variação das médias quadráticas das amplitudes, o que causaria grandes diferenças na percepção do volume sonoro e possivelmente no reconhecimento das diferentes sentenças.

O ajuste das médias quadráticas das amplitudes das sentenças do conjunto A a um valor alvo de -20 dBFS, sem a aplicação do vocoder, resulta em médias quadráticas das amplitudes com média e a mediana igual a -20 dBFS, com desvio padrão de apenas 0.01 dBFS.

Já para o mesmo conjunto das sentenças A, após aplicação do vocoder, o ajuste automático para um nível alvo de -20dBFS, resulta em média e mediana das amplitudes ajustadas de -20,34 dBFS e -20,31 dBFS respectivamente, com desvio padrão de 0,12 dBFS, e diferença máxima de 0,8 dBFS, conforme se

verifica na Figura 13. Nota-se portanto uma variação um pouco maior em sentenças que passaram pelo vocoder.

O conjunto B de arquivos também foi processado por todos os códigos em python, com exceção somente da etapa da vocoderização. Todas as sentenças deste conjunto também tiveram seus níveis de média quadrática de amplitudes ajustadas pelos códigos a um valor alvo de -20 dBFS. A Figura 14 representa um histograma dos níveis de média quadrática das amplitudes da pasta de 106 arquivos sonoros deste conjunto antes da aplicação do ajustes, evidenciado que os níveis médios quadráticos variam entre -28 dBFS e -14 dBFS, ainda que a maioria dos arquivos têm níveis médios quadráticos entre -20 dBFS e -14 dBFS. A Figura 15 mostra os níveis médios quadráticos da mesma seleção de arquivos após a modificação dos níveis médios quadráticos a um valor alvo de -20 dBFS. O histograma evidencia que o ajuste foi muito preciso, com a média dos valores RMS chegando a -20 dBFS, com mediana também de -20 dBFS e o desvio padrão de 0,04 dBFS.

Ao finalizar todas as etapas de processamento, todos os arquivos apresentam-se modificados. Estes, são salvos em formato .WAV com a taxa de amostragem em 44.100 Hz e RMS próximos ou iguais a 20 dBFS.

4. DISCUSSÃO

Os códigos de processamento de sinais de fala desenvolvidos na linguagem computacional Python foram avaliados frente a diferentes aspectos de qualidade. Em particular podem ser destacados a o processo de *fade-in* e *fade-out* exemplificado na Figura 5, que proporcionou uma transição suave de sons nas extremidades do sinal sonoro, não comprometendo a clareza da fala da sentença. Isto porque, o processo realizou a inserção da transição de forma gradual, essencial para evitar distorções sonoras.

A correta função do filtro passa-alta, com frequência de corte de 80 Hz, pode ser visualizada na comparação dos espectrogramas apresentados nas Figuras 6 e 7, que ilustram de forma figurada, a remoção de componentes espectrais de baixa frequência, para melhorar a clareza do sinal e contribuir com a função correta da etapa posterior de ajuste dos valores RMS. Ao ser implementado de maneira automatizada, o filtro foi aplicado em todas as

sentenças da mesma maneira, resultando na melhora da qualidade do áudio em todos os arquivos.

O ajuste de valores RMS é altamente eficiente. A partir dos histogramas apresentadas nas Figuras 11 e 12 verifica-se que a diferença entre os maiores níveis de amplitudes RMS e menores níveis de amplitudes RMS ajustadas não extrapola nunca 1 dB, para nenhuma das sentenças processadas, o que garante que a sensação de volume sonoro para todas os arquivos de áudio é a mesma, considerando que a diferença no limiar do observável para a sensação de volume sonoro é 1 dB¹⁷.

A simulação do processamento de áudio de um implante coclear através do vocoder nas sentenças é importante, pois permite que possamos reproduzir as condições de percepção auditivas dos usuários destes dispositivos. Além de permitir que haja a investigação do impacto que as diferentes configurações do processamento possuem nos resultados de reconhecimento de fala.

Para realizar os processos descritos acima desenvolvidos em Python, além da aplicação do Vocoder demora-se em média 40 minutos para todos os arquivos do conjunto A de 987 sentenças, em um notebook da marca Dell, modelo Inspiron 3583 com processador CORE i5 de oitava geração.

Futuramente pretende-se desenvolver ainda uma interface GUI para facilitar o uso dos códigos por pesquisadores não familiarizados com a linguagem computacional Python.

5. CONCLUSÃO

Tendo em vista a necessidade de dispor de uma ferramenta computacional para rápida modificação automatizada de grandes quantidades de arquivos de áudio, em particular gravações de sentenças, foram desenvolvidos códigos na linguagem computacional Python. Estes, não precisam ser usados exclusivamente da maneira apresentada por esta pesquisa, mas podem ser adaptados para serem usados de acordo com as necessidades de cada aplicação.

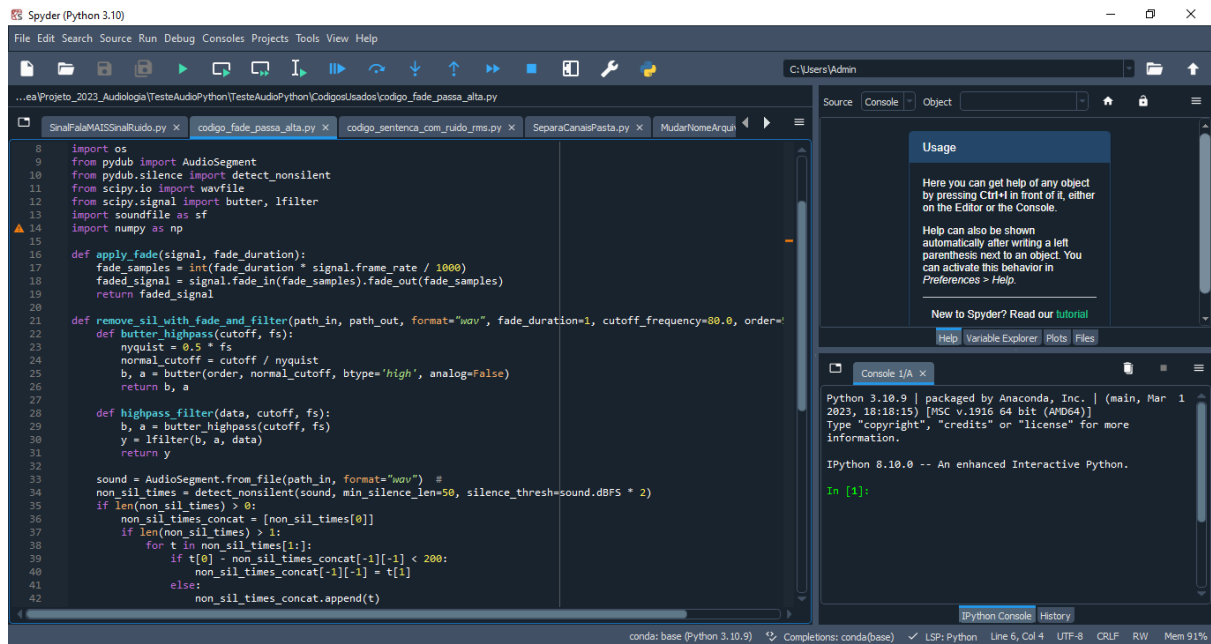
Tais códigos estão disponíveis no site GitHub, com acesso aberto à comunidade científica e com a possibilidade da realização de aprimoramento das técnicas apresentadas, nomeado de Códigos_Processamento_Sinal_De_Fala.

6. REFERÊNCIAS

1. BRASIL. Biblioteca Virtual em Saúde. Ministério da Saúde. **“Pra ouvir para a vida, escute com cuidado”**: 03/3 – Dia Mundial da Audição. 2022. Disponível em:
<https://bvsms.saude.gov.br/pr-ouvir-para-a-vida-escute-com-cuidado-03-3-dia-mundial-da-audicao/>. Acesso em: 13 jul. 2023.
2. Organização Mundial de Saúde (org.). **World report on hearing**. 2021. Disponível em: <https://www.who.int/publications/i/item/9789240020481>. Acesso em: 06 jun. 2024.
3. GENEBRA. Who. World Health Organization (org.). **WHO: 1 in 4 people projected to have hearing problems by 2050**. 2021. Disponível em: <https://www.who.int/news/item/02-03-2021-who-1-in-4-people-projected-to-have-hearing-problems-by-2050>. Acesso em: 13 jul. 2023.
4. COSTA, Lidiéli dalla et al. Teste de palavras no ruído: desenvolvimento, validação e valores de referência. **Codas**, [S.L.], v. 36, n. 3, p. 1-7, 2024. FapUNIFESP (SciELO). <http://dx.doi.org/10.1590/2317-1782/20242023091pt>.
5. SALVATO, Carolina de Campos et al. Correlação entre reconhecimento de fala, tempo de privação auditiva e tempo de uso de Implante Coclear em usuários com surdez pós-lingual. **Distúrbios da Comunicação**, [S.L.], v. 32, n. 3, p. 396-405, 3 set. 2020. Pontifical Catholic University of Sao Paulo (PUC-SP). <http://dx.doi.org/10.23925/2176-2724.2020v32i3p396-405>.
6. FERREIRA, Matheus Carvalho et al. Influence of different types of noise on sentence recognition in normally hearing adults. **Revista Cefac**, [S.L.], v. 23, n. 5, p. 1-9, out. 2021. FapUNIFESP (SciELO). <http://dx.doi.org/10.1590/1982-0216/20212352121>.
7. PINHEIRO, Maria Madalena Canina et al. Adaptação de listas de sentenças para avaliação da percepção da fala. **Codas**, [S.L.], v. 34, n. 1, p. 1-12 . 2022. FapUNIFESP (SciELO). <http://dx.doi.org/10.1590/2317-1782/20202020301>.
8. MURTA, Bernardo Henrique Pereira. **Plataforma para ensaios de percepção sonora com fontes distribuídas aplicável a dispositivos auditivos: perSONA**. 2019. 130 f. Dissertação (Mestrado) - Curso de Engenharia Mecânica, Universidade Federal de Santa Catarina, Florianópolis, 2019.

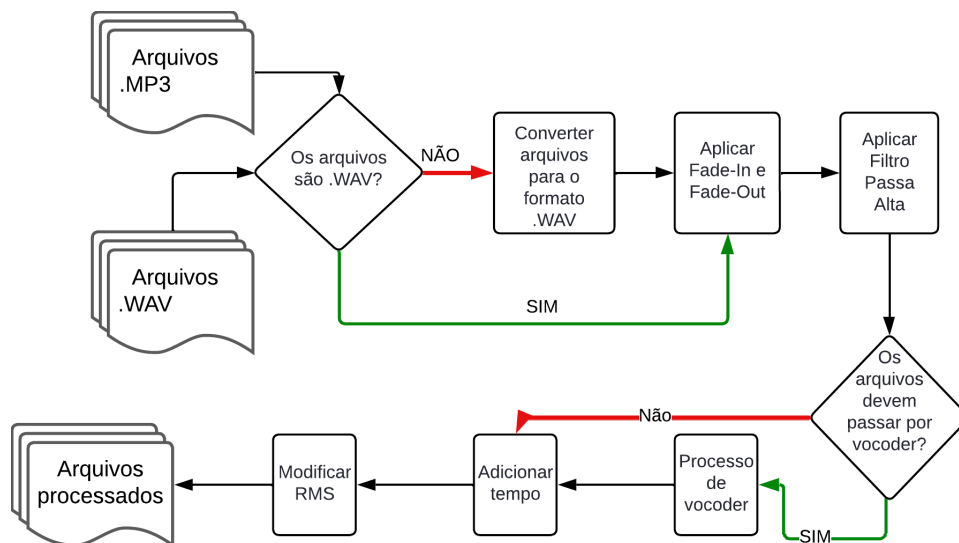
9. UPADHYAY, Navneet et al. A Multi-Band Speech Enhancement Algorithm Exploiting Iterative Processing for Enhancement of Single Channel Speech. *Journal Of Signal And Information Processing*, [S.L.], v. 04, n. 02, p. 197-211, 2013. Scientific Research Publishing, Inc.. <http://dx.doi.org/10.4236/jsip.2013.42027>.
10. KAROUI, Chadlia et al. Searching for the Sound of a Cochlear Implant: evaluation of different vocoder parameters by cochlear implant users with single-sided deafness. *Trends In Hearing*, [S.L.], v. 23, n. 1, p. 233121651986602, jan. 2019. SAGE Publications. <http://dx.doi.org/10.1177/2331216519866029>.
11. MASSA, Sean T. et al. Comparing the Performance Plateau in Adult Cochlear Implant Patients Using HINT and AzBio. *Otology & Neurotology*, [S.L.], v. 35, n. 4, p. 598-604, abr. 2014. Ovid Technologies (Wolters Kluwer Health). <http://dx.doi.org/10.1097/mao.0000000000000264>.
12. SPAHR, Anthony J. et al. Performance of Subjects Fit With the Advanced Bionics CII and Nucleus 3G Cochlear Implant Devices. *Archives Of Otolaryngology–Head & Neck Surgery*, [S.L.], v. 130, n. 5, p. 624, 1 maio 2004. American Medical Association (AMA). <http://dx.doi.org/10.1001/archotol.130.5.624>.
13. SPAHR, Anthony J. et al. Development and Validation of the AzBio Sentence Lists. *Ear & Hearing*, [S.L.], v. 33, n. 1, p. 112-117, jan. 2012. Ovid Technologies (Wolters Kluwer Health). <http://dx.doi.org/10.1097/aud.0b013e31822c2549>.
14. PYTHON SOFTWARE FOUNDATION. Documentação oficial da biblioteca padrão do Python. Disponível em: <https://docs.python.org/pt-br/3/library/index.html>. Acesso em: 16 dez. 2024.
15. PYTHON PACKAGE INDEX. Repositório de software para a linguagem de programação Python. Disponível em: <https://pypi.org/>. Acesso em: 18 dez. 2024.
16. FUCHS, Robert et al. The Effects of mp3 Compression on Acoustic Measurements of Fundamental Frequency and Pitch Range. *Speech Prosody 2016*, [S.L.], v. [], n. [], p. 523-527, 31 maio 2016. ISCA. <http://dx.doi.org/10.21437/speechprosody.2016-107>.
17. FORINASH, Kyle et al. **Sound - An Interactive eBook**. [S.L.]: Libretexts Platform, 2024. 281 p.

Figura 1 – Exemplo de Código no Software Spyder para Modificação de Sentenças.



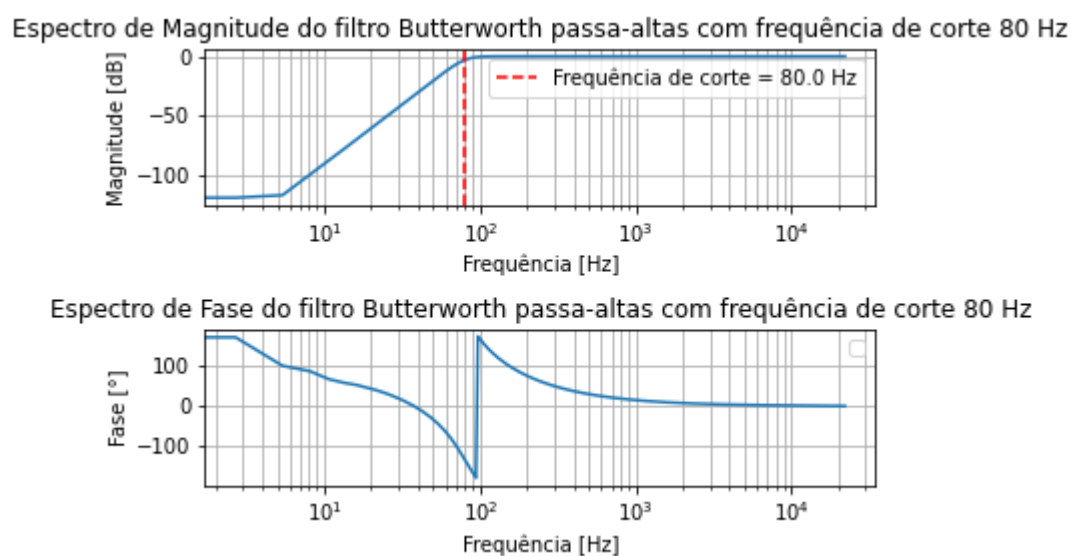
Fonte: Elaborada pelo autor, 2023.

Figura 2 – Fluxograma do Processo de Modificação de Sentenças de Áudio.



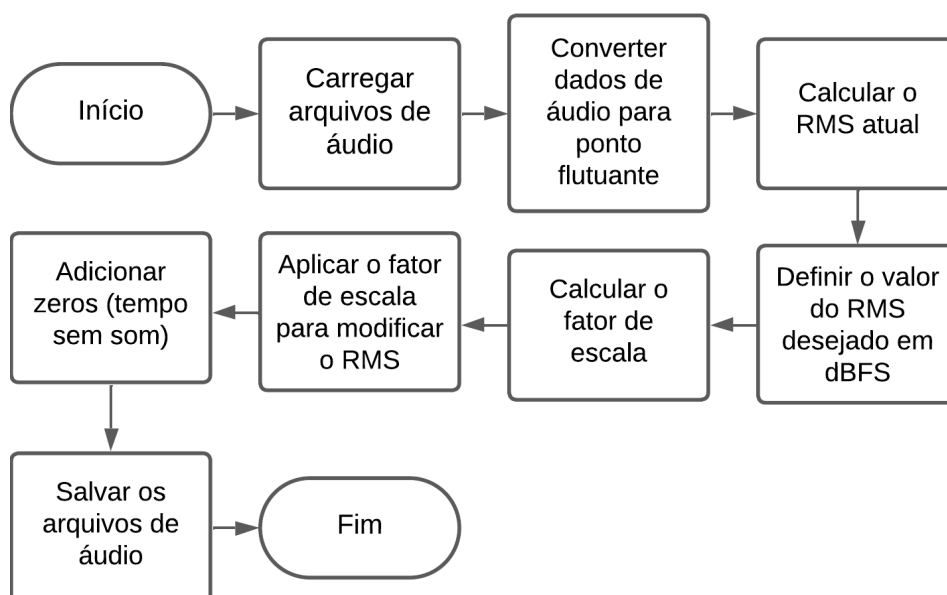
Fonte: Elaborada pelo autor, 2023.

Figura 3 – Espectro de magnitude e fase do filtro passa-alta com frequência de corte em 80 Hz.



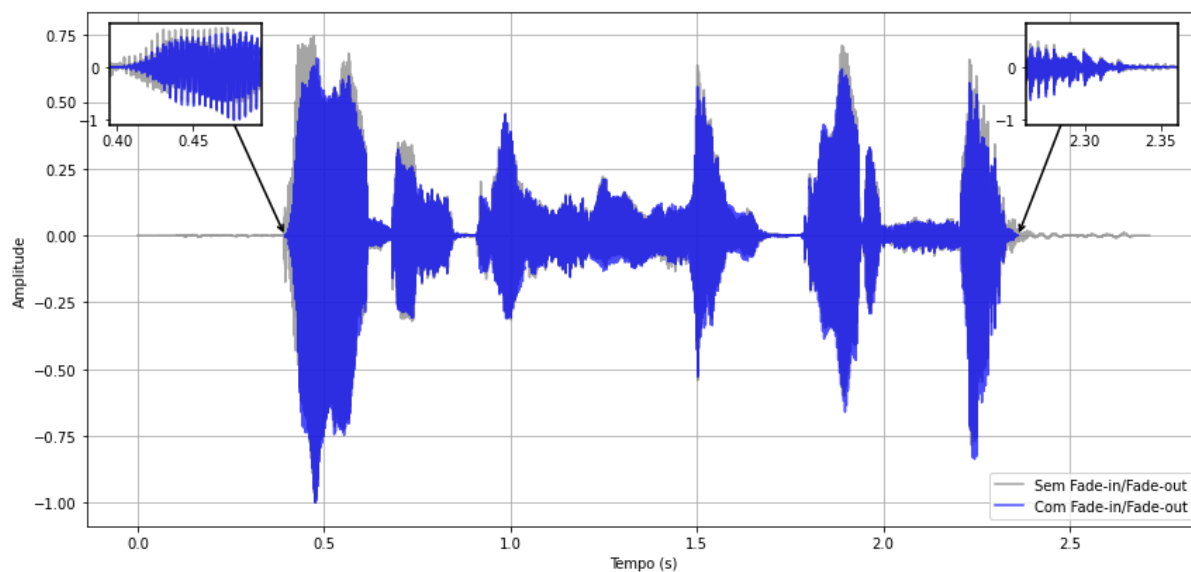
Fonte: Elaborada pelo autor, 2024.

Figura 4 – Fluxograma do Processo de Ajuste de RMS em sinais de áudio.



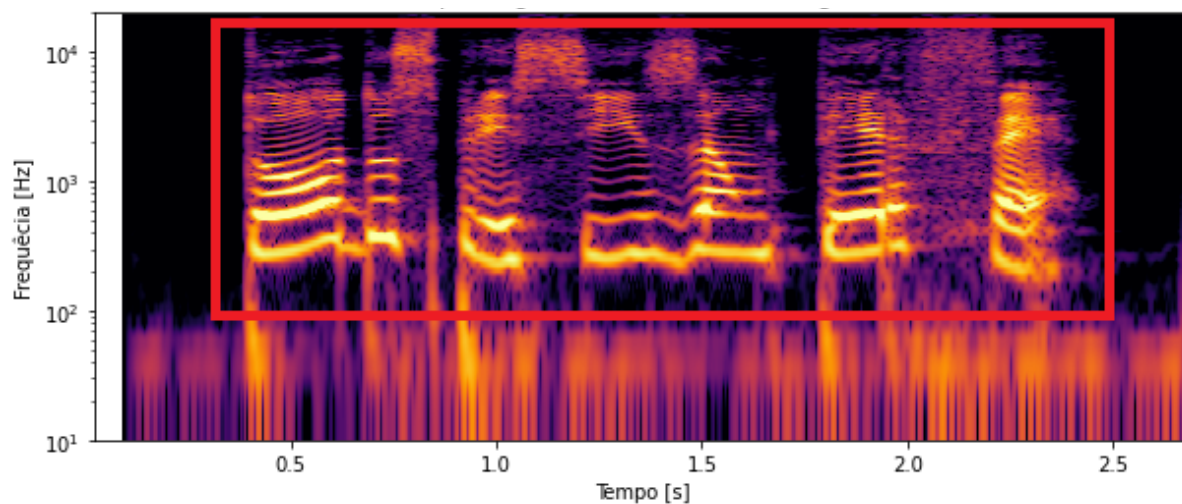
Fonte: Elaborada pelo autor, 2023.

Figura 5 – Histórico no tempo do sinal sonoro antes do *Fade-In/Out*, Remoção de Silêncio e Filtro Passa-Alta e depois.



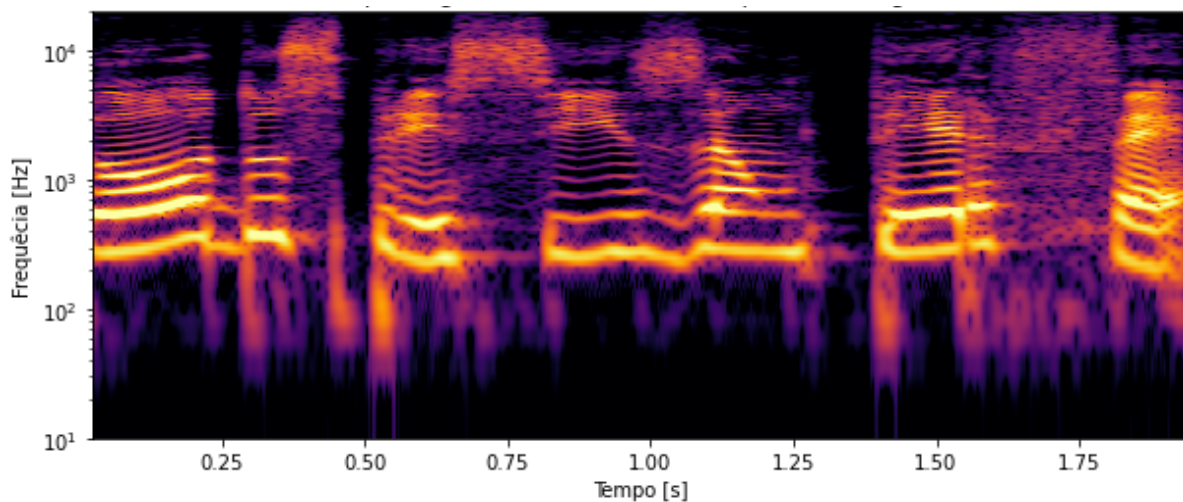
Fonte: Elaborada pelo autor, 2023.

Figura 6 – Espectrograma do sinal sonoro antes do *Fade-In/Out*, Remoção de Silêncio e Filtro Passa-Alta.



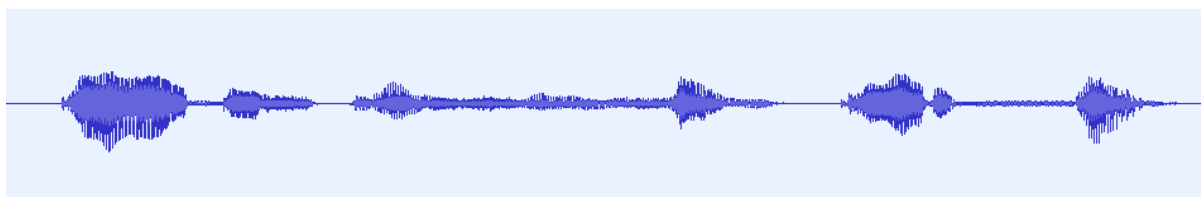
Fonte: Elaborada pelo autor, 2023.

Figura 7 – Espectrograma do sinal sonoro após aplicação do *Fade-In/Out*, Remoção de Silêncio e Filtro Passa-Alta.



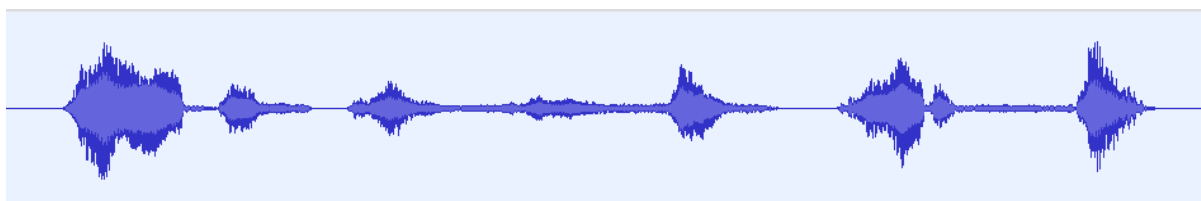
Fonte: Elaborada pelo autor, 2023.

Figura 8 – Histórico no tempo de um arquivo de áudio exemplo antes do processo de vocoderização por vocoder tonal de 5 canais.



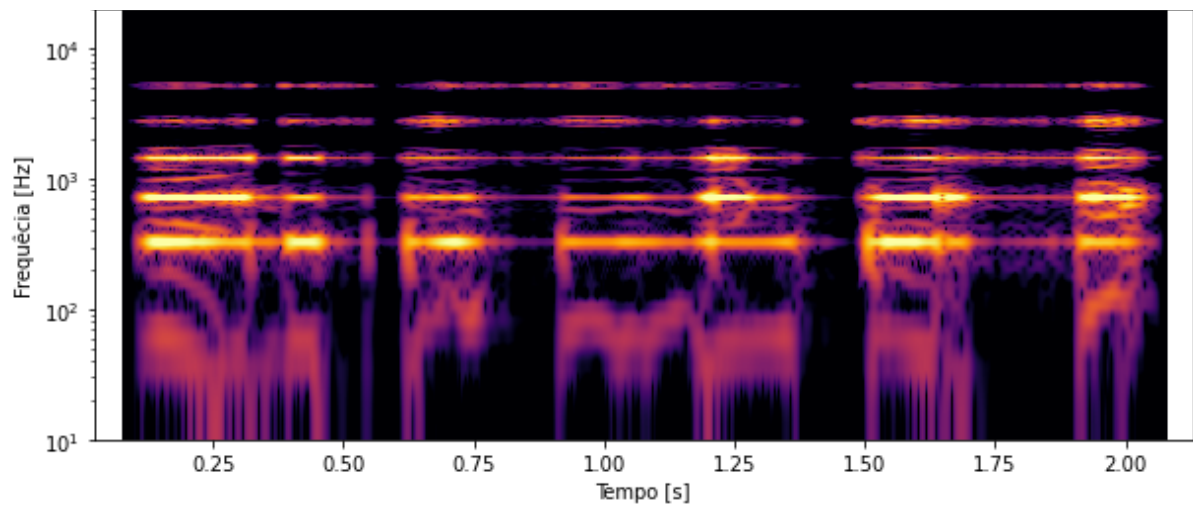
Fonte: Elaborada pelo autor, 2023.

Figura 9 – Histórico no tempo de um arquivo de áudio exemplo após o processo de vocoderização por vocoder tonal de 5 canais.



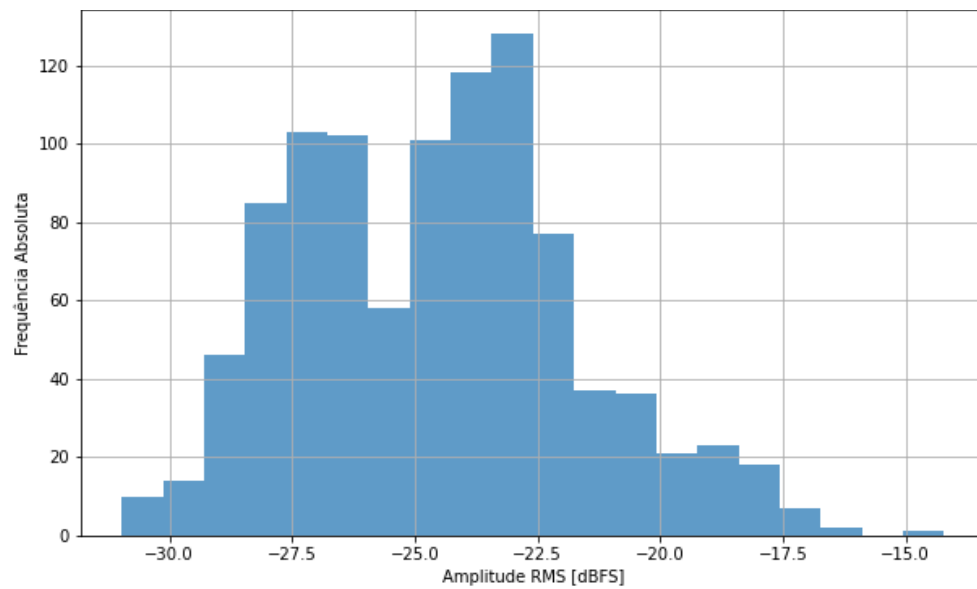
Fonte: Elaborada pelo autor, 2023.

Figura 10 – Espectrograma do sinal sonoro após processamento com vocoder tonal de 5 canais.



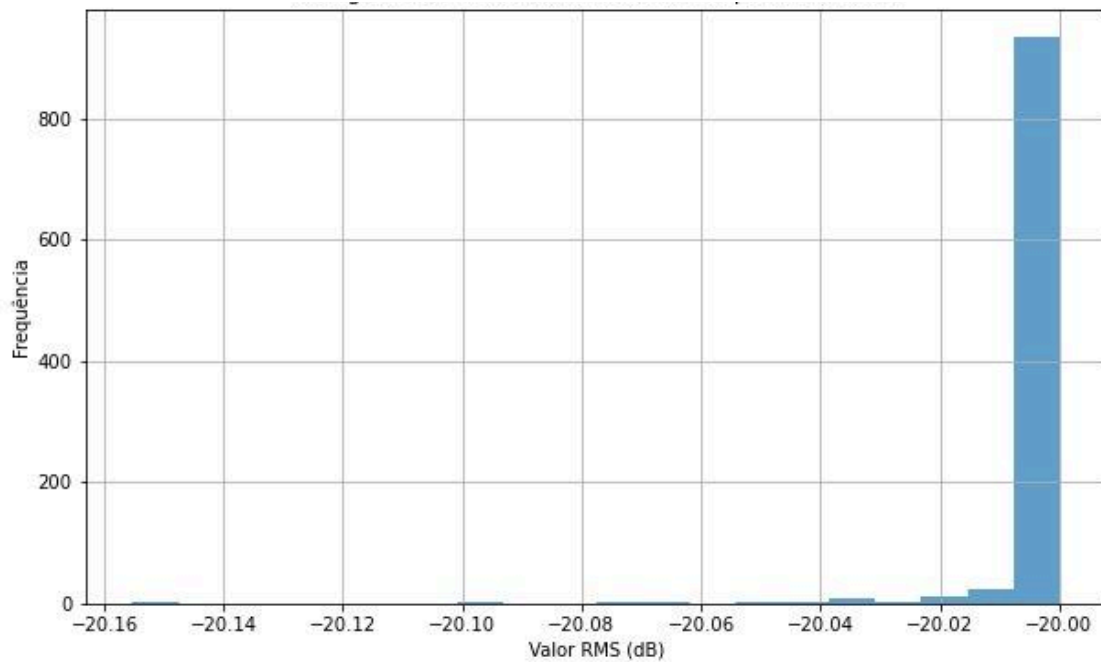
Fonte: Elaborada pelo autor, 2024.

Figura 11 – Histograma da distribuição de valores RMS do conjunto A de 987 sentenças não processadas pelo ajuste de médias quadráticas.



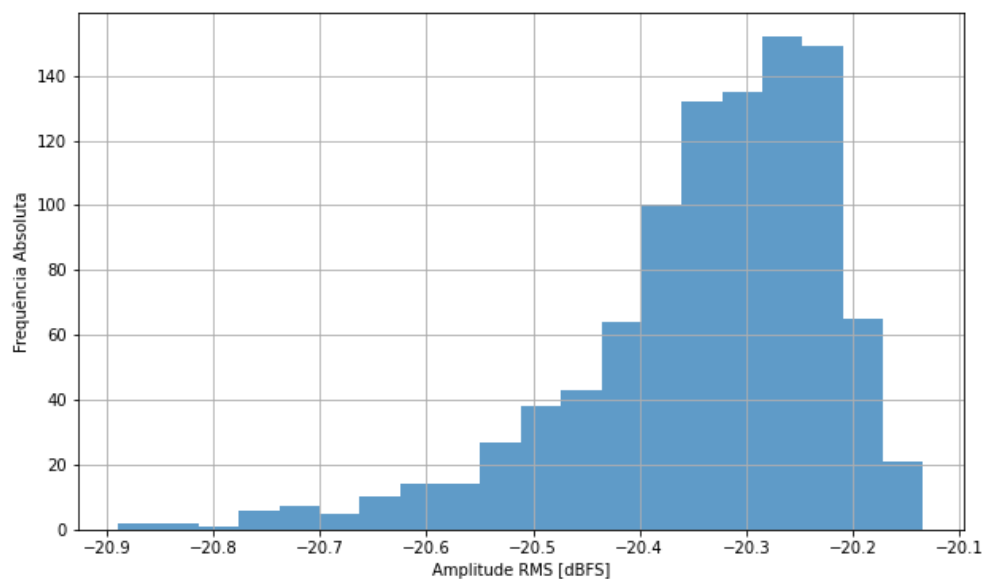
Fonte: Elaborada pelo autor, 2023.

Figura 12 – Histograma da distribuição dos níveis das amplitudes RMS do conjunto A de 987 sentenças com ajuste do RMS para o valor alvo de -20dBFS e sem o vocoder.



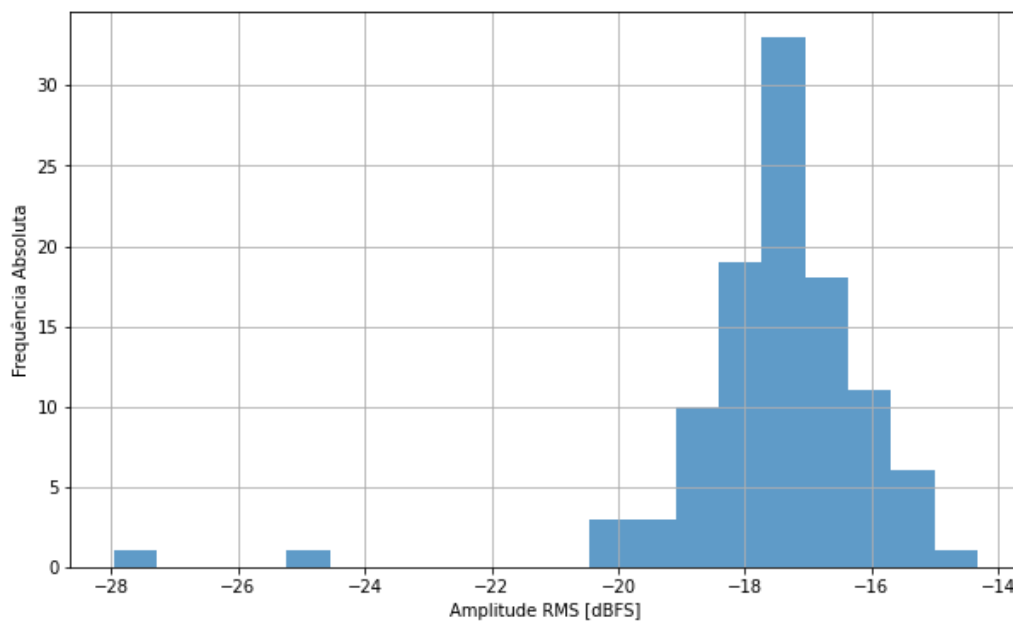
Fonte: Elaborada pelo autor, 2024.

Figura 13 – Histograma da distribuição dos níveis das amplitudes RMS do conjunto A de 987 sentenças após o vocoder e ajuste do RMS para o valor alvo de -20 dBFS



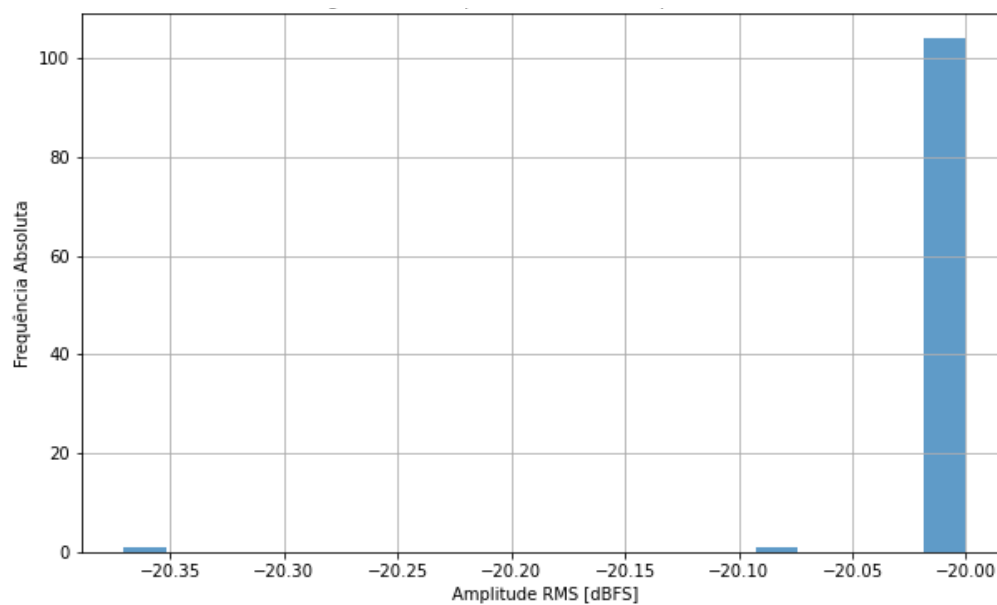
Fonte: Elaborada pelo autor, 2023.

Figura 14 – Histograma da distribuição de valores RMS do conjunto B de 106 sentenças não processadas.



Fonte: Elaborada pelo autor, 2023.

Figura 15 – Histograma da distribuição dos níveis das amplitudes RMS do conjunto B de 106 sentenças após o ajuste do RMS para o valor alvo de -20dBFS.



Fonte: Elaborada pelo autor, 2023.