UNIVERSIDADE FEDERAL DO MARANHÃO

Programa de Pós-Graduação em Ciência da Computação

Bianca Valéria Lopes Pereira

Reconhecimento de fonemas com compactação das frequências via centroide e redes stacked autoencoders

São Luís
2024

BIANCA VALÉRIA LOPES PEREIRA

RECONHECIMENTO DE FONEMAS COM COMPACTAÇÃO DAS FREQUÊNCIAS VIA CENTROIDE E REDES *STACKED AUTOENCODERS*

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Areolino de Almeida Neto

SÃO LUÍS - MA
2024

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Diretoria Integrada de Bibliotecas/UFMA

Pereira, Bianca Valéria Lopes.
Reconhecimento de Fonemas Com Compactação das Frequências Via Centroide e Redes Stacked Autoencoders / Bianca Valéria Lopes Pereira. - 2024.
114 f.

Orientador(a): Areolino de Almeida Neto.
Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luís, 2024.

1. Reconhecimento de Fonemas. 2. Coeficientes de Compactação. 3. Stacked Autoencoders. 4. . 5. . I. Almeida Neto, Areolino de. II. Título.

BIANCA VALÉRIA LOPES PEREIRA

RECONHECIMENTO DE FONEMAS COM COMPACTAÇÃO DAS FREQUÊNCIAS VIA CENTROIDE E REDES *STACKED AUTOENCODERS*

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Aprovada em 06 de junho de 2024

Prof. Dr. Areolino de Almeida Neto
Universidade Federal do Maranhão
Orientador

Prof. Dr. Alexandre César Muniz de Oliveira
Universidade Federal do Maranhão
Examinador Interno

Prof. Dr. Nelson Cruz Sampaio Neto
Universidade Federal do Pará
Examinador Externo

São Luís - MA
2024

Dedico a YHWH, aos meus pais, aos que já sofreram ou sofrem com a CID 10 - F31 e F41.1 e a todas as meninas, garotas e mulheres que optaram pelas ciências exatas e engenharias.

AGRADECIMENTOS

Agradeço a YHWH pela vida que Ele me concedeu, por promover saúde e forças para chegar até o final dessa etapa. Sem Ele nada seria possível.

Aos meus pais João Cristóvão Pereira Filho e Aurineia S. Lopes Pereira, que estiveram ao meu lado. Esta dissertação é só mais uma prova de que todos os esforços deles pela minha educação não foram em vão e valeram a pena.

Aos meus avós maternos Mariano Lopes (*in memoriam*) e Maria dos Anjos. Aos meus avós paternos, Raimunda e João Cristovam (*in memoriam*).

Ao meu irmão Davi, que fique gravado que a minha conquista também é tua. Sou grata à toda a minha família pelo apoio que sempre me deram durante toda a minha vida.

Aos amigos que ao longo dessa caminhada lançaram palavras positivas sobre a minha vida, por todas as orações e pela cumplicidade.

Ao meu orientador, professor Areolino de Almeida Neto, pelo incentivo, palavras de apoio nos momentos difíceis, palavras de conforto nos momentos de lágrimas, sabedoria, pela dedicação do seu tempo ao meu projeto de pesquisa e pela experiência compartilhada de vida.

Aos colegas do Laboratório InovTec, Rômulo e Francisco, por toda ajuda e suporte durante a pesquisa. Gratidão pelos momentos de risadas.

A minha orientadora de monografia e amiga Marta de Oliveira Barreiros. Quem me incentivou a ingressar no PPGCC, com quem compartilhei minhas angustias e dúvidas e ela tão sábia soube me ouvir, mas jamais deixou eu desistir em momento algum desse título. Obrigada por acreditar em mim, que eu iria conseguir vencer mais essa etapa e por toda empatia.

Por fim, agradeço aqueles que serviram como referência para os resultados desse trabalho.

“Ainda estou longe de ser o que quero, mas com
a ajuda de Deus terei sucesso.”
(Vicent Van Gogh)

RESUMO

O reconhecimento de fonemas é uma área da linguística e processamento de fala que envolve identificar e distinguir os sons distintivos que compõem uma língua. Reconhecer fonemas envolve a capacidade de discernir e categorizar os diferentes sons da fala, mesmo quando há variações de pronúncia, contexto ou entonação. Neste trabalho, é proposto um modelo de reconhecimento de fonemas utilizando uma rede *stacked autoencoder*, denominada CollabNet. A CollabNet introduz um método colaborativo para inserção de novas camadas escondidas, em contraste com o tradicional empilhamento de autoencoders. Na CollabNet, a adição de uma nova camada é feita de forma coordenada e gradual, permitindo ao projetista controlar sua influência no treinamento. Essa colaboração garante que o aprendizado da nova camada se integre de forma eficaz com as camadas anteriores, resultando em um treinamento mais alinhado e eficiente. Para a representação dos fonemas, foi realizada a compactação das frequências por meio de centroides, de maneira que se preserve as particularidades do som. Com o objetivo de criar uma representação geométrica dos áudios das bases de dados, foi calculada a transformada rápida de Fourier (FFT) para cada amostra de áudio, em seguida foram agrupadas as frequências e foi calculado o centroide de cada grupo. Posteriormente, a rede *deep stacked autoencoder* foi parametrizada e treinada para o reconhecimento de sílabas fonemas. Com essa representação dos áudios, foi possível manter sua caracterização particular de maneira que a CollabNet identificasse os diversos sons da língua portuguesa do Brasil, tendo assim uma acurácia de 75,96% e PER de 23,73%.

Palavras-chave: Reconhecimento de fonemas. Coeficientes de compactação. *Stacked autoencoders*.

ABSTRACT

Phoneme recognition is an area of linguistics and speech processing that involves identifying and distinguishing the distinctive sounds that make up a language. Recognizing phonemes involves the ability to discern and categorize the different sounds of speech, even when there are variations in pronunciation, context or intonation. In this work, a phoneme recognition model is proposed using a *stacked autoencoder* network, called CollabNet. CollabNet introduces a collaborative method for inserting new hidden layers, in contrast to the traditional stacking of autoencoders. In CollabNet, the addition of a new layer is done in a coordinated and gradual manner, allowing the designer to control its influence on the training. This collaboration ensures that the learning of the new layer is effectively integrated with the previous layers, resulting in more aligned and efficient training. To represent the phonemes, the frequencies were compacted using centroids so as to preserve the particularities of the sound. In order to create a geometric representation of the audios in the databases, the fast Fourier transform (FFT) was calculated for each audio sample, then the frequencies were grouped and the centroid of each group was calculated. Subsequently, the *deep stacked autoencoder* network was parameterized and trained to recognize phonetic syllables. With this representation of the audios, one could maintain their particular characterization so that CollabNet could identify the various sounds of the Brazilian Portuguese language, thus achieving an accuracy of 75.96% and a PER of 23.73%.

Keywords: Phoneme recognition. Compaction coefficients. Stacked autoencoders.

LISTA DE ILUSTRAÇÕES

LISTA DE ABREVIATURAS E SIGLAS

ACU	Acurácia
AFI	Alfabeto Fonético Internacional
ASR	Reconhecimento automático de falante, em inglês
ASV	Verificação automática de falantes, em inglês
BM	Máquinas de Boltzmann, em inglês
CNNs	Redes neurais convolucionais, em inglês
CTC	Modelo de classificação temporal conexionista
DNNs	Redes neurais profundas, em inglês
DBNs	Redes de crenças profundas, em inglês
DMBs	Máquinas profundas de Boltzmann, em inglês
DL	*Deep learning*
DSN	*Stacked autoencoder* tradicional
EBM	Modelos baseados em energia, em inglês
FBANKs	Banco de filtros de registro de Mel, em inglês
FSC	F-Score
FST	Transdutores de estados finitos
GB	Gigabyte
HMM	Modelos ocultos de Markov, em inglês
HTK	Kit de ferramentas de modelo oculto de Markov, em inglês
IA	Inteligencia artificial
LPC	Coeficientes de predição linear, em inglês
LPREFC	Coeficientes de reflexão de predição linear, em inglês
MELSPEC	Coeficientes do banco de filtros de Mel, em inglês
MFCC	Coeficientes cepstrais de frequência de Mel, em inglês
ML	*Machine learning*
MLP	Perceptron multicamadas, em inglês
MSE	Erro quadrático médio, em inglês
NARX	Rede auto regressiva não linear com entradas exógenas, em inglês
PER	Taxa de erro de fonemas, em inglês

PLP	Predição linear perceptiva, em inglês
PPCA	Análise de componentes principais probabilística, em inglês
PRE	Precisão
QCNNs	Redes neurais convolucionais com valores de quatérnio, em inglês
RAM	Memória de Acesso Aleatório, em inglês
RCIL	Reconhecimento de comando independente do falante, em inglês
RNAs	Redes neurais artificiais
RNNs	Redes neurais recorrentes, em inglês
RBMs	Máquinas restritas de Boltzmann, em inglês
SEN	Sensibilidade
SOM	Mapas auto-organizáveis, em inglês
STFT	Transformada de Fourier de curta duração, em inglês
SVM	Máquinas de vetores de suporte, em inglês
WT	Transformada de Wavelets, em inglês

SUMÁRIO

1 INTRODUÇÃO

A utilização de aparelhos eletrônicos como computadores, *smartphones*, *smartwatches*, entre outros, está se tornando cada vez mais difundida em nossa rotina. A computação ubíqua, também conhecida como computação pervasiva, é um conceito que descreve a integração natural e invisível da tecnologia em nosso cotidiano. Nesse paradigma, os dispositivos eletrônicos estão presentes em diversos objetos do nosso dia a dia, como celulares, relógios, eletrodomésticos e até mesmo roupas. Esses dispositivos são capazes de comunicar-se entre si, compartilhar informações e adaptar-se às nossas necessidades, proporcionando uma experiência tecnológica contínua e fluida.

A computação ubíqua busca transformar a maneira como interage-se com o mundo, tornando-o mais conectado, eficiente e conveniente, ao permitir que a tecnologia esteja sempre ao alcance de seus usuários, de forma discreta e integrada ao estilo de vida destes. Antes mesmo dessa integração, sempre houve uma preocupação constante no campo da informática com a forma de interação e comunicação com esses dispositivos. Com a computação ubíqua, isso se acentuou.

Nos primeiros aplicativos computacionais, as interações eram realizadas exclusivamente por meio do teclado, utilizando linhas de comando e blocos de texto. Ao longo dos anos, houve uma evolução em direção a interações mais intuitivas, com o surgimento das interfaces gráficas, que ofereceram maior facilidade na manipulação das máquinas. Nos dias atuais, é cada vez mais comum o uso de *softwares* que operam diretamente por comandos de voz. A voz é um aspecto dos seres humanos e está fortemente relacionada às necessidades de agrupamento e comunicação (SOLTI, 2022).

O reconhecimento de padrões é um campo fascinante da ciência da computação que envolve a capacidade de identificar regularidades e estruturas em conjuntos de dados complexos. Este busca desenvolver algoritmos e técnicas que permitem aos computadores aprender com exemplos e reconhecer automaticamente padrões em diferentes domínios, como imagens, áudio, texto e dados numéricos.

O reconhecimento de padrões tem aplicações amplas e diversas, desde a detecção de fraudes em transações financeiras até o reconhecimento facial, passando pela análise de sinais biomédicos e previsão de tendências em dados de mercado. Ao explorar os dados e extrair informações significativas, o reconhecimento de padrões desempenha um papel fundamental no avanço da inteligência artificial e impulsiona o desenvolvimento de sistemas mais inteligentes e automatizados.

O reconhecimento de padrões é uma capacidade própria dos seres humanos. Entre os vários modelos que podem ser identificados, existe o reconhecimento de voz (CARVALHO, 2020). O reconhecimento de padrões via eletrônicos abrange técnicas de concessão de padrões às suas relativas especificações ou estereótipos. Esse procedimento, deve ocorrer por máquinas

e com a mínima interferência humana possível (GONZALEZ; WOODS, 2009). Contudo, a atividade de reconhecimento não é facilmente reproduzida por uma máquina digital.

Para o reconhecimento de voz, em princípio, executa-se uma fase de pré-processamento nos áudios, que possibilita extrair suas particularidades, reduzir ruídos e as dessemelhanças entre falantes. Posteriormente, é efetuada uma etapa de treinamento e classificação, aplicando algoritmos de *machine learning* com a finalidade de identificar os fonemas (DIJKSTRA, 2021).

As pesquisas iniciais nessa área tinham como propósito reconhecer um sinal sonoro e transfigurá-lo em algarismos (FURUI, 1981). Depois deste princípio, inúmeros protótipos foram executados. Na década de 80, alguns pesquisadores dessa área começaram a aplicar a teoria probabilística dos modelos ocultos de Markov (HMM) em sistema de reconhecimento de fala e deram início aos primeiros modelos comerciais de reconhecimento de fala (RABINER, 1989). Na década de 90, modelos comerciais tornaram-se afamados, tendo como exemplos o *IBM Via Voice* e o *Dragon Systems* (CONIAM, 1999).

A tecnologia inicial de reconhecimento de fala se baseava na análise de espectrograma e na audição humana. O espectrograma é uma representação das várias frequências de um determinado sinal ao longo do tempo. No entanto, tais métodos são muito subjetivos e contam com especialistas bem treinados para identificá-los corretamente (OLIVEIRA NETO *et al.*, 2010).

Nos dias atuais, as técnicas mais utilizadas para a representação do som são os *Mel-Frequency Cepstral Coefficients* (MFCC) e os *Logarithmic Mel-Filter Bank Coefficients* (FBANKs) (MEFTAH *et al.*, 2016 apud CARVALHO, 2020; GROENBROEK, 2021). A utilização de técnicas de pré-processamento tem o objetivo de aprimorar a extração de características representativas, que desempenham um papel crucial no processo de reconhecimento de padrões. Isso direciona os esforços do modelo para compreender as informações realmente relevantes, enquanto elimina as demais, como ruídos de fundo ou espectros sem expressividade (CARVALHO, 2020).

Os sistemas de reconhecimento de voz podem ser divididos em três grupos: reconhecimento automático de falante (ASR), verificação automática de falantes (ASV) e o reconhecimento de comando independente do falante (RCIL). No ASR, o sistema deve ser capaz de reconhecer diferentes vozes em um grupo fixo de indivíduos. Já o ASV verifica apenas se a voz capturada está autorizada, isto é, se o locutor está registrado. E no RCIL, o sistema pode identificar as palavras, não tendo importância quem as fala (NIQUINI, 2007 apud OLIVEIRA NETO *et al.*, 2010).

Deste modo, implementar uma abordagem de recognição de voz, mais precisamente na perspectiva de RCIL, para identificar fonemas foi o desafio deste trabalho. Isto foi realizado utilizando centroides como representações de áudio e aplicando a técnica de redes neurais artificiais (RNAs). Esta abordagem possibilita ao usuário uma comunicação onidirecional com a máquina. Assim, foi construído um modelo baseado em redes neurais, do tipo *stacked autoencoder*, para identificação de sílabas fonéticas. Também foi comparado o modelo produzido com outros

modelos presentes na literatura.

1.1 OBJETIVOS

O objetivo geral desta pesquisa foi desenvolver uma abordagem para a identificação de sílabas fonéticas, fazendo uso de representações de áudio por meio de centroides e aplicando a técnica de redes neurais profundas do tipo *stacked autoencoders* para o reconhecimento de padrões.

Os objetivos específicos foram:

1. Desenvolver técnica de compactação/representação/modelagem de sinais fonéticos baseada em cálculo de centroide;
2. Implementar uma rede neural artificial profunda do tipo *stacked autoencoder* com inserção colaborativa de camadas para reconhecimento de sílabas fonéticas;
3. Obter uma base de áudio de português do Brasil com dados saneados;
4. Comparar o modelo produzido com outros modelos presentes na literatura.

1.2 JUSTIFICATIVAS E MOTIVAÇÕES

A principal motivação para pesquisar e desenvolver soluções na área de reconhecimento de voz é possibilitar aos usuários uma comunicação mais natural com a máquina. A interação por meio de áudio elimina a necessidade de contato físico ou visual, tornando a comunicação mais acessível e conveniente.

Com o avanço da tecnologia, novas aplicações tornaram-se mais próximas dos usuários comuns. O reconhecimento de voz é amplamente utilizado devido à sua capacidade de estabelecer uma comunicação direta entre um usuário e um programa ou dispositivo. Isso permite que tanto usuários leigos como aqueles com dificuldades visuais possam usufruir de certas aplicações. Desta forma, para existir a linguagem verbal por meio da fala e consequentemente a comunicação, o reconhecimento por voz é de suma importância.

Um aspecto fundamental a ser considerado é a inclusão educacional, pois muitas pessoas não possuem habilidades suficientes de escrita, mas ainda assim podem se comunicar por meio da fala ou de gestos. Nesse sentido, o reconhecimento de voz permite uma interação homem-máquina inclusiva, atendendo também às necessidades de pessoas com deficiências.

Para exemplificar esta utilidade, têm-se aplicativos de jogos, como exemplo o Duolingo e o Pou, ou aplicativos de redes sociais, como Youtube e WhatsApp, que conseguem atender às crianças, jovens, adultos e idosos.

O Youtube exibe os vídeos requisitados como resposta de pesquisa por voz. Em casos onde a criança ainda não necessariamente saiba ler ou escrever ou o idoso que ainda não enfrentou os todos os obstáculos da inclusão digital, ainda sim, ambos entendem o ícone de voz, que é o

microfone, e pode simplesmente apertá-lo e falar o que se deseja ver, tendo assim a interação por voz.

O Youtube então retorna possíveis resultados e basta um clique para consumir o conteúdo. Não precisa de treinamento ou um aprendizado por parte do usuário, o simples algoritmo reconhece as palavras e transformam-nas em itens para fazer uma busca e exibir seus resultados.

Da mesma forma é o WhatsApp, que possibilita a comunicação entre dois ou mais usuários por meio de áudio, ligação, vídeo chamada e também tem o recurso do microfone, onde o usuário clica nele, fala e ele faz o reconhecimento dos *tokens* ou palavras e escreve o que foi falado.

As vantagens da interação homem-computador por áudio são numerosas, incluindo uma comunicação mais eficiente, sem a necessidade de contato visual ou físico. Além disso, possibilita a inclusão digital de pessoas com baixa escolaridade e melhora a qualidade de vida de indivíduos com limitações motoras, permitindo-lhes utilizar dispositivos digitais por meio de comandos de voz.

Os exemplos do reconhecimento de voz são inúmeros e podem ajudar a melhorar as tarefas e interações humanas. É possível adquirir assistentes virtuais, como exemplo: a Cortana, a Siri, a Alexa, entre outras. Elas executam tarefas simples, mas estas tarefas podem ser ampliadas cada vez mais, podendo chegar a um nível onde muitas interações poderão ser automatizadas. Um exemplo seria a Alexa atender uma compra em um restaurante realizada por comando de voz, sem o contato direto do cliente com alguma superfície, o que deveria ser evitado ao máximo durante à crise da Covid-19.

Embora existam diversas soluções de reconhecimento de fala e comandos por voz atualmente, o desenvolvimento contínuo de novas tecnologias nessa área é de extrema importância. Muitas das soluções existentes ainda apresentam erros de tradução de áudio para texto e exigem conexão com a internet, já que seus algoritmos são baseados em servidores em nuvem. Portanto, aprimorar essas tecnologias pode levar a uma maior eficiência e disponibilidade para os usuários.

1.3 TRABALHOS RELACIONADOS

Esta seção expõe os trabalhos relacionados e recentes da área de processamento de voz e uso de técnicas de *machine learning* para reconhecimento de fonemas. Ao se aplicar alguma técnica no áudio para criar uma representação de dados costuma apresentar bons resultados. Isso ocorre devido aos dados de áudio brutos apresentarem uma alta quantidade de informações, sendo assim, a aplicação de pré-tratamentos ou utilização de métodos de extração de características reduz o volume de informações a serem identificados.

O artigo de Petry *et al.* (1999) apresentou técnicas de pré-processamento, extração de parâmetros para detecção de padrões e utilizou um algoritmo de quantização vetorial que a partir de uma base de dados inicial verifica o locutor da mensagem. Eles utilizaram o algoritmo de Luft (1994) para detecção de limites da palavra, posteriormente a janela de Hamming e por fim

extraíram os coeficientes cepstrais.

O trabalho de Matuck e SILVA (2005) apresentou a importância da implementação da transformada discreta de Fourier nas amostras de voz anteriormente ao processamento das mesmas por redes neurais. As propriedades da transformada aplicada no sinal do áudio tornaram expressiva a melhora no processamento.

No estudo conduzido por Ding *et al.* (2015), foi realizado um experimento comparativo entre os extratores FBANKs, MFCC e *Linear Prediction Coefficients* (LPC). Os modelos acústicos gerados pelos extratores de áudio foram utilizados como entrada para uma *Deep Belief Network* (DBN). Durante o experimento, observou-se que o modelo acústico baseado em FBANKs apresentou um desempenho superior em relação aos demais.

No estudo realizado por Meftah *et al.* (2016), foi conduzida uma análise dos principais extratores de características utilizados na literatura, utilizando um modelo baseado em HMM como classificador. Esse estudo se concentrou em uma base de fonemas da língua árabe. Foram comparados os seguintes extratores: LPC, MFCC, *Perceptual Linear Prediction* (PLP), FBANKs, *Mel-Filter Bank Coefficients* (MELSPEC) e *Linear Prediction Reflection Coefficients* (LPREFC). O sistema que alcançou a melhor acurácia foi baseado em FBANKs, seguido pelo segundo melhor sistema, que utilizou espectrograma na escala de Mel (MELSPEC). Os estudos também comprovaram que MFCC e PLP apresentaram acurácia semelhante, enquanto LPC não obteve bons resultados na tarefa de reconhecimento de fonemas.

Segundo Parcollet *et al.* (2018) o modelo de classificação temporal conexionista (CTC) acoplado a redes neurais recorrentes (RNN) ou redes neurais convolucionais (CNN) tornou mais fácil o treinamento de sistemas de reconhecimento de fala de ponta a ponta. No entanto, em modelos de valores reais, componentes temporais, como energias de bancos de filtros mel e coeficientes cepstrais obtidos a partir deles, juntamente com suas primeiras e segundas derivadas, são processados como elementos individuais, enquanto uma alternativa natural é processar esses componentes como entidades compostas.

Seguindo essa linha, Parcollet *et al.* (2018) propôs agrupar esses elementos na forma de quaternions e processar esses quaternions usando a álgebra de quaternions. Foi proposta essa integração de múltiplas visualizações de características em redes neurais convolucionais com valores de quatérnio (QCNN), a ser usada para mapeamento de sequência para sequência com o modelo CTC. Resultados promissores são relatados usando QCNNs simples em experimentos de reconhecimento de fonemas com o corpus *TIMIT*. Mais precisamente, QCNNs obtêm uma taxa de erro de fonemas (PER) mais baixa com menos parâmetros de aprendizado do que um modelo concorrente baseado em CNNs de valores reais.

O artigo de Catete e Rosa (2023) teve como objetivo utilizar a tecnologia de reconhecimento de fala para realizar uma avaliação probabilística, analisando a variação da taxa de acertos e acurácia dos fonemas sustentados de interesse, bem como as variações inerentes à ferramenta utilizada. O foco foi dado aos casos de pessoas entre 45 e 60 anos, divididos em grupos de indivíduos com fala normal e indivíduos com fala disfônica, que foram utilizados para treinar e

testar o sistema de reconhecimento. A etiquetagem manual foi realizada para todo o conjunto de sinais utilizando a ferramenta Praat, enquanto o reconhecimento de fala foi realizado por meio de HMM da ferramenta *Hidden Markov Model Toolkit* (HTK).

A utilização de técnicas para processamento de áudio com o objetivo de criar uma representação de dados é geralmente eficaz, pois os dados de áudio brutos contêm uma grande quantidade de informações. Nesse sentido, a aplicação de pré-processamentos ou a utilização de métodos de extração de características permite reduzir o volume de informações a serem identificadas.

Tendo isso em vista, trabalhos recentes dividem a tarefa de reconhecimento de áudio em dois módulos, o primeiro módulo busca elaborar um modelo acústico, o qual é responsável por extrair características representativas do espectro de áudio. E então, essas características são posteriormente enviadas a um segundo módulo, que é composto por um classificador, o qual deverá correlacionar as informações de forma a identificar as sequências analisadas.

O trabalho de Valiati (2000) teve como objetivo o desenvolvimento de uma aplicação de RNAs, com a arquitetura *multi-layer perceptron* (MLP), capaz de reconhecer um vocabulário restrito de comandos pronunciados isoladamente e que independe do locutor. Os áudios foram gravados no formato *wav*, os mesmos foram pré-processados, onde cada amostra foi submetida a um processo de normalização do sinal e posteriormente a detecção do seu início e fim, assim eliminando os ruídos. Para determinar os limiares, foi utilizado o algoritmo proposto por Rabiner e Sambur (1975). Em seguida, foram extraídas as informações relevantes do sinal que caracterizem o som, utilizando a janela de Hamming e coeficientes cepstrais (Cepstrum).

O trabalho de Oliveira Neto *et al.* (2010) teve como objetivo implementar um sistema de reconhecimento de comandos com locutores restritos em um pequeno vocabulário. O sistema deveria ser capaz de identificar qual a palavra pronunciada pelo locutor e, para isso, esse trabalho utilizou várias técnicas conjugadas, a fim de minimizar ruídos e interferências. Para a caracterização dos sinais, foram utilizados os coeficientes cepstrais, os LPCs e os coeficientes de um modelo autorregressivo (AR). Para a identificação e para a interpretação dos padrões, foram utilizadas RNAs, com a utilização da arquitetura MLP.

O artigo de Kamarudin *et al.* (2014) apresentou o processo de identificação automática da pronuncia do Alcorão. As técnicas de extração de características utilizadas para a identificação das regras dos versos do Alcorão/Tajweed incluem os MFCC, que são propensos a ruído aditivo e podem reduzir o resultado da classificação. Por conseguinte, para melhorar o desempenho do MFCC, Kamarudin *et al.* (2014) adicionaram as características do centroide espectral proposto para serem usadas na extração de características dos acentos do Alcorão.

O centroide espectral é uma medida utilizada no processamento digital de sinais para caraterizar um espectro. Esse centroide indica onde se situa um suposto centro de massa do espectro. Perceptualmente, tem uma ligação forte com a impressão de brilho de um som (GREY; GORDON, 1978), sendo por vezes designado por centro de massa espectral (PULAVARTI *et al.*, 2022).

É calculado, conforme a Equação 1.1, como a média ponderada das frequências presentes no sinal, determinada usando a transformada de Fourier, com as suas magnitudes como pesos (PEETERS, 2004).

$$Centroid = \frac{\sum_{n=0}^{N-1} f(n)x(n)}{\sum_{n=0}^{N-1} x(n)} \tag{1.1}$$

Fonte: (KAMARUDIN *et al.*, 2014)

onde $x(n)$ representa o valor da magnitude da posição n e $f(n)$ representa a frequência central dessa posição.

Ao implementarem o recurso do centroide espectral, Kamarudin *et al.* (2014) complementaram a melhoria da precisão na identificação dos acentos do Alcorão. O algoritmo de classificação de padrões utilizado faz uso da técnica de redução dimensional da Análise de Componentes Principais Probabilística (PPCA) nas características e o Modelo de Mistura Gaussiana, com o propósito de modelar a eficácia da combinação de extração de características.

O trabalho de Zhang *et al.* (2017) propôs uma estrutura de fala ponta a ponta na qual combinaram CNNs com a abordagem CTC sem camadas recorrentes intermediárias. Como as CNNs são adeptas da modelagem de estruturas locais nas entradas, eles utilizaram os FBANKs com deltas e delta-deltas que preservam as correlações locais do espectrograma.

Vários trabalhos propuseram modelos acústicos usando a combinação de algumas técnicas de transformação e técnicas de aprendizagem profunda. No trabalho de Quintanilha *et al.* (2017), foram utilizadas CNN e RNNs. O sinal de áudio de um trecho de fala é transformado em MFCCs e usado como entrada do sistema. Eles usaram o conjunto de dados de fala do português brasileiro (BRSD), que foi construído pela combinação de 4 conjuntos de dados de diferentes fontes (LapsBM, CSLU: Spoltech Brazilian Portuguese, Voxforge e Sid dataset).

O artigo de Cardona *et al.* (2017) apresentou como proposta uma identificação online dos fonemas da língua portuguesa por meio de um modelo auto regressivo não linear com entradas exógenas, comumente denominado NARX. As redes NARX são consideradas redes neurais dinâmicas com retardo de tempo, exceto pela restrição de que apenas o vetor de resultado da camada de saída é realimentado como uma entrada retardada.

Além disso, para Cardona *et al.* (2017) avaliar as melhorias na precisão do reconhecimento para cada categoria consonantal (oclusivas, fricativas, nasais, africadas e líquidas), bem como para cada categoria vocálica (orais, ditongos e nasais). Cardona *et al.* (2017) também calculou e estudou a taxa média de precisão para cada uma dessas categorias. Para isto, foi explorada as arquiteturas da MLP em conjunto com a rede neural dinâmica para o reconhecimento dos fonemas da língua portuguesa.

A pesquisa de Aouani e Ben (2018) utilizou uma rede neural do tipo *autoencoder* para classificação de emoções a partir de dados de voz. O autor extraiu os atributos a partir dos dados de áudio utilizando MFCC que busca mudanças temporais bruscas no espectro de voz. A partir da saída do MFCC, os autores utilizaram a rede neural do tipo *autoencoder* para o reconhecimento

dos padrões e por fim usaram uma máquina de vetores de suporte (SVM) para a classificação das emoções.

No trabalho de CARVALHO (2020), foi empregado um detector chamado *Single Shot Detection* em conjunto com a arquitetura de rede convolucional MobileNet. Dois conjuntos de dados compostos por áudios na língua inglesa foram utilizados. Para cada amostra de áudio, os espectrogramas foram calculados na escala de Mel, fornecendo sua representação gráfica. A fim de treinar o algoritmo de detecção de localização de fonemas, a posição temporal de cada ocorrência de fonema foi anotada em seus respectivos espectrogramas.

CARVALHO (2020) utilizou dois modelos com arquiteturas diferentes foram empregados: o modelo MobileNet - *Large* e o modelo MobileNet - *Small*. Os treinamentos utilizando os dados após a aplicação das técnicas de deslocamento, alteração de velocidade e adição de ruído localizado apresentaram um índice mAP superior ao treinamento utilizando apenas os dados originais do conjunto de dados.

No trabalho de Lee *et al.* (2021), uma membrana artificial foi projetada e fabricada. Ela produziu um padrão de deslocamento espacial em resposta a um sinal audível, que foi utilizado para treinar uma CNN. Nos modelos de valor real, os componentes do intervalo de tempo, como energias do banco de filtro mel e os coeficientes cepstrais obtidos a partir deles, juntamente com seus derivados de primeira e segunda ordem, são processados como elementos individuais, enquanto uma alternativa natural é processar tais componentes como entidades compostas.

O trabalho de Slaviero (2022) teve como objetivo avaliar o uso da rede SincNet em um modelo de aprendizado profundo para extrair características relevantes do sinal de áudio, a fim de realizar a identificação da língua em músicas. Além disso, foram empregadas diferentes técnicas de processamento de sinais para eliminar informações irrelevantes (como som instrumental ou ruídos da plateia) do sinal de música.

O sistema proposto por Slaviero (2022) realizou, primeiramente, a segmentação dos segmentos onde a voz cantante está presente, seguida pela separação do sinal de voz do som instrumental. O sinal de voz foi então alimentado na rede de aprendizado profundo para a extração de características e a identificação da língua. O sistema proposto foi avaliado em um conjunto de dados construído a partir de músicas de um serviço de streaming. Os resultados mostraram que as etapas de pré-processamento, segmentação e separação contribuíram significativamente para o desempenho do sistema.

O artigo de Nedjah *et al.* (2023) apresentou o uso de redes neurais *ensemble* para o reconhecimento automático da fala de fonemas da língua portuguesa. O processo se iniciou com um pré-processamento do sinal de fala para extrair as principais características do mesmo num determinado instante de tempo.

Inspirados pelo princípio “dividir e conquistar”, Nedjah *et al.* (2023) superaram a complexidade do reconhecimento automático de fala criando modelos baseados em um conjunto de redes neurais especialistas, permitindo dividir o vasto espaço de decisão relacionado ao reconhecimento de fala de modo que cada rede se ocupe apenas com uma área delimitada desse

espaço de decisão. A aplicação dessa estratégia melhorou a precisão, a sensibilidade e a exatidão do processo de reconhecimento.

Cada rede especialista incluída no conjunto decidiu sobre cada uma das amostras de entrada pré-processadas. O conjunto de decisões obtido é ponderado. Portanto, a rede especialista com o maior peso para a saída determinou a classificação final da amostra. Depois disso, Nedjah *et al.* (2023) executaram uma etapa de pós-processamento dinâmico, implementada como uma rede neural recorrente. O objetivo foi atenuar o efeito oscilatório que ocorre durante o reconhecimento de classes com características semelhantes. Dois conjuntos foram investigados. O primeiro era baseado no agrupamento de classes fonéticas semelhantes, enquanto o segundo cuidava da distribuição desequilibrada de amostras no conjunto de treinamento.

À luz do meu conhecimento, ainda não existe alguma proposta na literatura que aborde a representação de som via centroide (centro geométrico de figura plana) após aplicada a transformada rápida de Fourier. Assim também não existe na literatura algum trabalho que utilize a CollabNet para reconhecimento de fonemas.

Por fim, o trabalho a ser citado é sobre a extração de centroide espectral através da transformada de Wavelet Packet. Roque e Mendes (2013) propuseram uma nova técnica de extração do descritor tempo-frequencial centroide espectral a partir da transformada Wavelet Packet. Roque e Mendes (2013) afirmaram que ainda são necessários mais estudos comparativos entre as duas técnicas, principalmente na análise de outros tipos de sinais, e a respeito da complexidade do cálculo. A partir do centroide espectral wavelet novos descritores podem ser criados seguindo o mesmo conceito, como o espalhamento espectral e qualquer outro descritor baseado em momentos espectrais.

1.4 CONTRIBUIÇÕES

As principais contribuições deste trabalho são:

- Representar o som por meio do centroide (centro geométrico de uma figura plana);
- Obter modelo de reconhecimento de sílabas fonéticas utilizando redes *stacked autoencoders* com inserção colaborativa;
- Comparar o desempenho do modelo de identificação de fonemas com outros modelos propostos na literatura.

1.5 ORGANIZAÇÃO DO TRABALHO

O trabalho está dividido em sete capítulos. Além deste, no Capítulo 2, é apresentada a fundamentação teórica sobre som, voz, fala, língua, fonema e sinal. Esse capítulo visa, ainda, a assegurar o entendimento dos conceitos que são abordados na área de reconhecimento de fonemas. No Capítulo 3, são apresentados os conceitos sobre redes neurais artificiais, tendo

como foco as redes *deep feedfoward*, além dos tipos de redes *deep autoencoders*. No Capítulo 4, são apresentados os materiais e as ferramentas que foram empregadas na construção do modelo proposto. No Capítulo 5, são apresentadas a CollabNet e suas definições. O Capítulo 6 aborda os resultados dos experimentos realizados e as discussões decorrentes. O Capítulo 7 apresenta a conclusão, enfatizando os resultados alcançados e delineando possíveis direções futuras de pesquisa.

2 PROCESSAMENTO DE VOZ

Este capítulo define o que é som, voz, fala, língua, fonema e sinal. A Seção 2.1 apresenta o conceito de voz, língua, letras, alfabeto e o ponto de vista da sociolinguística com relação aos fenômenos linguísticos e sociais, os conceito de fonética e fonologia e o alfabeto fonético internacional. Na Seção 2.2, é explicado o som do ponto de vista ondulatório e suas várias áreas de estudo. Na Seção 2.3, são apresentadas as várias transformações que comumente são ou podem ser aplicadas nos áudios depois de gravado e armazenado digitalmente.

2.1 VOZ E FALA

A voz é uma característica inata dos seres humanos que se baseia na habilidade de produzir sons articulados e possibilita estabelecer a comunicação entre os indivíduos, desde que os mesmos falem e compreendam a mesma língua (CARVALHO, 2020).

A voz está associada ao som produzido pela vibração das pregas vocais, enquanto a fala é o som articulado na boca através da ação dos articuladores do som. A fala está diretamente ligada às palavras. A voz e a fala estão interligadas e caminham juntas (SLAVIERO, 2022).

A língua é um sistema de representação que consiste em palavras e regras que se combinam para expressar ideias, processos, sentimentos e declarações concisas e abrangentes que permitem a comunicação. Essas regras e essas palavras são membros da comunidade, então a língua pertence a cada comunidade (WEISS *et al.*, 2018).

O livro de Cunha e Cintra (2016) afirma que uma língua é um sistema gramatical pertencente a um grupo de indivíduos. Ela é o recurso pois ela concebe o universo que a cerca e sobre ele se comporta. Ela é uma utilização social da faculdade da linguagem, criação da sociedade, não pode ser imutável; ao contrário, tem de viver em perpétua evolução, paralela à do organismo social que a criou.

Uma língua é constituída de um conjunto infinito de frases. Cada uma delas possui uma face sonora, ou seja a cadeia falada, e uma face significativa, que corresponde ao seu conteúdo. Uma frase, por sua vez, pode ser dividida em unidades menores de som e significado, as quais se denominam palavras, e em unidades ainda menores, que apresentam apenas a face sonora, os fonemas. As palavras são unidades menores que a frase e maiores que o fonema (CUNHA; CINTRA, 2016).

Para serem reproduzidas na escrita as palavras de uma língua, é empregado um certo número de sinais gráficos chamados de letras. O conjunto ordenado das letras de que são usadas para transcrever os sons do idioma falado denomina-se alfabeto (CUNHA; CINTRA, 2016). Cada idioma possui seu próprio conjunto de caracteres (CARVALHO, 2020), como apresentado na Tabela 1.

Tabela 1 – Exemplos de alfabetos em alguns idiomas.

Língua	Alfabeto	Alfabeto estendido
Portuguesa	a b c d e f g h i j k l m n o p q r s t u v w x y z	à á â ã ç é ê í ó ô õ ú
Inglesa	a b c d e f g h i j k l m n o p q r s t u v w x y z	
Espanhola	a b c d e f g h i j k l m n o p q r s t u v x y z	á é í ñ ó ú ü
Galesa	a b c d e f g h i l m n o p r s t u w y	ŵ ŷ
Grega	$\alpha\beta\gamma\delta\varepsilon\zeta\eta\theta\iota\kappa\lambda\mu\nu\xi$ **o** $\pi\rho\sigma\tau\upsilon\phi\chi\psi\omega$	

Fonte: Elaborado pela autora (2023).

Com o desenvolvimento da sociolinguística, ramo da linguística que estuda a língua como fenômeno social e cultural, mostrou-se que as relações entre a língua e a sociedade passaram a ser caracterizadas com maior precisão. observa-se uma covariação entre os fenômenos linguísticos e sociais, ou seja, eles estão interligados e se influenciam mutuamente. No entanto, em certos casos, é mais apropriado considerar uma relação direcional, ou seja, a influência da sociedade sobre a língua ou da língua sobre a sociedade (MEILLET, 1926 apud CUNHA; CINTRA, 2016).

A concepção da língua como um instrumento de comunicação social, flexível e diversificado em todos os seus aspectos, é uma noção relativamente recente. Ela é vista como um meio de expressão para indivíduos que vivem em sociedades que também são diversas em termos sociais, culturais e geográficos. Nesse sentido, uma língua histórica não é um sistema linguístico único, mas sim um conjunto de sistemas linguísticos inter-relacionados, formando um diassistema. Essa perspectiva implica que o estudo de uma língua se torna extremamente complexo e requer uma delimitação precisa dos fenômenos analisados para controlar as variáveis que influenciam os diversos níveis e eixos de diferenciação. A variação sistemática agora é amplamente incorporada tanto na teoria quanto na descrição da língua (CUNHA; CINTRA, 2016).

Em sua essência, uma língua possui, pelo menos três tipos de diferenças internas, que podem variar em sua profundidade:

- Diferenças geográficas, também conhecidas como variações diatópicas, que se referem às variações linguísticas entre diferentes regiões geográficas, incluindo dialetos locais, variantes regionais e até mesmo variações intercontinentais.

- Diferenças entre as diferentes camadas socioculturais, chamadas variações diastráticas, que envolvem variações linguísticas entre diferentes grupos sociais e culturais, como o nível culto, a língua padrão, o nível popular, entre outros.

- Diferenças entre os diferentes tipos de modalidade expressiva, conhecidas como variações diafásicas, que se referem às variações linguísticas entre diferentes formas de expressão, tais como a língua falada, a língua escrita, a língua literária, as linguagens específicas de determinadas áreas ou profissões, a linguagem utilizada por homens e mulheres, e assim por diante.

A partir da nova concepção da língua como diassistema, tornou-se possível esclarecer numerosos casos de polimorfismo, pluralidade de normas e a inter-relação de fatores geográficos, históricos, sociais e psicológicos que influenciam o funcionamento complexo de uma língua e direcionam sua evolução.

Condicionada de maneira consistente dentro de cada grupo social e parte integrante da competência linguística de seus membros, a variação é, portanto, inerente ao sistema da língua e ocorre em todos os níveis: fonético, fonológico, morfológico, sintático, entre outros. Essa multiplicidade de manifestações do sistema não compromete suas capacidades funcionais (CUNHA; CINTRA, 2016).

Todas as variações linguísticas possuem estrutura e correspondem a sistemas e subsistemas adaptados às necessidades de seus usuários. No entanto, a estreita conexão da lingua com a estrutura social e os sistemas de valores da sociedade resulta em uma avaliação distinta das características de suas diferentes modalidades diatópicas, diastráticas e diafásicas (CUNHA; CINTRA, 2016).

A língua padrão, por exemplo, mesmo sendo apenas uma das muitas variedades de um idioma, é sempre a mais prestigiosa, pois atua como modelo, norma e ideal linguístico para uma comunidade. Devido ao seu valor normativo, ela exerce uma influência coercitiva sobre as outras variedades, tornando-se uma força significativa contra a variação linguística (CUNHA; CINTRA, 2016).

As formas características que uma língua assume regionalmente são conhecidas como dialetos. No entanto, alguns linguistas fazem uma distinção específica entre as variedades diatópicas, ou seja, diferenciam o falar e o dialeto (CUNHA; CINTRA, 2016). Todavia, devido à dificuldade de caracterizar efetivamente as duas modalidades diatópicas, o termo “dialeto” é utilizado no sentido de variedade regional da língua, sem levar em conta seu maior ou menor distanciamento em relação à língua padrão.

Quando o objeto é a voz, uma das primeiras características que a definem é a língua. A língua é um primeiro passo para explorar mais pontos do seu âmbito, como o RCIL, o ASV e o ASR (CAMPBELL, 1997 apud SLAVIERO, 2022; NIQUINI, 2007 apud NETO *et al*., 2010).

Na frase “minha tia estava lavando louça na pia”, as palavras “tia” e “pia” distinguem-se apenas pelo elemento consonântico inicial. Qualquer diferença significativa entre duas palavras de uma língua estabelecida através da oposição ou contraste entre dois sons revela que cada um desses sons representa uma unidade mental sonora distinta. Essa unidade, da qual o som é a representação ou realização física, é denominada fonema.

Os sons consonânticos que diferenciam as palavras tia e pia correspondem, portanto, a diferentes fonemas. A disciplina que estuda detalhadamente os sons da fala, bem como as diversas realizações dos fonemas, é chamada de fonética. A parte da gramática que investiga o comportamento dos fonemas em uma língua é denominada fonologia, fonemática ou fonêmica.

2.1.1 Alfabeto Fonético Internacional e transcrição fonética e fonológica.

Existe o Alfabeto Fonético Internacional (AFI) e é por meio dele que as línguas estabelecem o seu alfabeto fonético. O AFI é um sistema padronizado utilizado para representar os sons da fala de todas as línguas humanas. Desenvolvido no final do século XIX e continuamente atualizado e aprimorado ao longo dos anos, o AFI se tornou uma ferramenta essencial para linguistas, fonoaudiólogos, professores de línguas e outros profissionais que trabalham com a análise e descrição dos sons da fala.

A necessidade de um sistema fonético internacional surgiu devido à grande diversidade de sons que existem nas línguas do mundo. Cada língua tem seu próprio conjunto de sons e muitas vezes um mesmo símbolo ou letra pode representar diferentes sons em línguas distintas. Isso cria desafios na comunicação e no estudo comparativo das línguas (SOLTI, 2022).

O AFI busca solucionar esse problema ao fornecer um conjunto de símbolos fonéticos, cada um representando um som específico e invariável. Cada símbolo do AFI é cercado por colchetes para indicar que se refere ao som e não à letra em si. Por exemplo, o som da vogal "a"em "casa"é representado pelo símbolo [a], enquanto o som da vogal "a"em "maçã"é representado pelo símbolo [ã].

Além de vogais, o AFI também cobre as consoantes, sons articulados com diferentes configurações dos órgãos da fala, como a língua, os lábios e os dentes. Por exemplo, o som "p"em "pato"é representado pelo símbolo [p], o som "m"em "maçã"é representado pelo símbolo [m], e assim por diante.

Outro exemplo, desta vez com uma palavra: [’kaw], pronúncia popular carioca, [’kal], pronúncia portuguesa normal e brasileira do Rio Grande do Sul, para a palavra sempre escrita cal (CUNHA; CINTRA, 2016).

Já os fonemas se transcrevem entre barras oblíquas: / /. Por exemplo: o fonema /s/ pode ser representado ortograficamente por s, como em saco; por ss, como em osso; por c, como em cera; por ç, como em poço; por x, como em próximo; e pode ser realizado como [s], no português normal de Portugal e do Brasil.

O AFI é uma ferramenta valiosa para o estudo comparativo das línguas, permitindo que linguistas e pesquisadores analisem e comparem os sons em diferentes idiomas de forma precisa e sistemática. Além disso, é uma ferramenta útil para o ensino de línguas estrangeiras, ajudando os alunos a compreender e reproduzir os sons corretamente.

O uso do AFI não se limita apenas ao campo acadêmico. Ele também é amplamente utilizado em áreas como a fonoaudiologia e o treinamento de locutores, auxiliando no tratamento de distúrbios de fala e no aprimoramento da pronúncia em diferentes idiomas (SOLTI, 2022).

Em suma, o Alfabeto Fonético Internacional (AFI) é uma conquista notável na área da linguística, proporcionando uma ferramenta valiosa e unificada para a análise e descrição dos sons da fala em todas as línguas. Sua contribuição para o estudo das línguas e da comunicação humana é inestimável, tornando-se um recurso indispensável para profissionais e estudiosos em

todo o mundo.

2.2 PROCESSAMENTO DIGITAL DE SINAIS

O som é uma vibração que se propaga pelo ar, transmitindo energia, mas sem transportar matéria. A velocidade do som pode variar de acordo com o meio em que a onda sonora se propaga. O som é considerado um sinal (CAMASTRA; VINCIARELLI, 2015).

Tanto na área da comunicação e da linguística, um sinal é qualquer meio utilizado para transmitir informações de um emissor para um receptor. O som é uma forma de sinal que ocorre quando as ondas sonoras são produzidas por uma fonte sonora (como a voz humana, instrumentos musicais, objetos em movimento, etc.) e se propagam através do ar ou de outros meios até atingir o ouvido de um receptor (CAMASTRA; VINCIARELLI, 2015; CUNHA; CINTRA, 2016).

Essas ondas sonoras carregam informações e podem representar diferentes significados, seja através da fala, da música, dos sons ambientais, entre outros. Quando as ondas sonoras alcançam o ouvido humano, elas fazem com que o tímpano vibre. Essas vibrações são então transmitidas ao ouvido interno, onde são convertidas em sinais elétricos que são enviados ao cérebro através do nervo auditivo. O cérebro do receptor processa e interpreta esses sinais, permitindo-nos ouvir e reconhecer diferentes sons e suas fontes. Com isso, o som cumpre sua função de transmitir mensagens, emoções, ideias ou qualquer outro tipo de conteúdo (CAMASTRA; VINCIARELLI, 2015).

O estudo do som é uma área de pesquisa conhecida como acústica e sua compreensão é essencial para diversas disciplinas, como a música, a fonoaudiologia, a engenharia de som e a ciência da comunicação.

Para fins didáticos, os sinais são categorizados em diversos aspectos, como a periodicidade, a energia, a potência, se são contínuos ou discretos no tempo, analógicos ou digitais, e se são determinísticos ou probabilísticos. Quando se refere aos termos “contínuo no tempo” e “discreto no tempo”, essas qualificações se relacionam com a representação do sinal ao longo do eixo de tempo, normalmente representado no eixo horizontal (CAMASTRA; VINCIARELLI, 2015; CARVALHO, 2020).

Por outro lado, os termos “analógico” e “digital” qualificam a amplitude do sinal, normalmente representada no eixo vertical. A fala é um sinal analógico, contínuo e não periódico. No entanto, durante a tarefa de aquisição do sinal de voz, as ondas de pressão emitidas pelo locutor são transmitidas através do ar e, em seguida, convertidas em ondas elétricas pelos microfones e amplificadores (CAMASTRA; VINCIARELLI, 2015; CARVALHO, 2020).

Devido a limitações técnicas, os computadores digitais não podem armazenar um sinal considerando um conjunto infinito de pontos, o que requer amostragem periódica. É crucial manter uma taxa de amostragem suficientemente alta para possibilitar a reconstrução do sinal original sem perda significativa de informação. O teorema da amostragem fornece o fundamento quantitativo necessário para esse propósito (LATHI, 2006; CAMASTRA; VINCIARELLI, 2015).

2.3 TRANSFORMAÇÕES DE SINAIS

Depois de gravado e armazenado digitalmente, é comum realizar algumas transformações no sinal. As transformações no sinal são procedimentos essenciais na área de processamento de sinais, onde os dados coletados ou gerados são submetidos a modificações para aprimorar a qualidade, facilitar a análise ou extrair informações relevantes.

Essas transformações podem incluir operações matemáticas como filtragem, convolução, modulação, e também técnicas avançadas como a transformada de Fourier, que converte o sinal do domínio do tempo para o domínio da frequência, revelando padrões espectrais ocultos.

Além disso, outras técnicas de processamento, como a normalização, a equalização e a compressão, são aplicadas para melhorar o desempenho do sinal em diversas aplicações, abrangendo desde comunicações sem fio e processamento de áudio e vídeo até aplicações médicas e científicas.

A escolha adequada das transformações no sinal é de extrema importância para otimizar a eficiência do processamento e obter resultados precisos e significativos a partir dos dados digitais obtidos. Nesta seção, mencionaremos algumas das transformações de sinais mais conhecidas.

2.3.1 Pré-enfase

A pré-enfase é uma técnica de processamento de sinais amplamente utilizada para melhorar a qualidade e a clareza de áudios transmitidos. Essa transformação consiste em realçar as frequências mais altas do sinal de áudio antes de sua transmissão ou armazenamento.

O processo de pré-enfase é realizado através da aplicação de um filtro passa-alta, que enfatiza as altas frequências e reduz a amplitude das baixas frequências. A aplicação do filtro de pré-ênfase a um sinal x pode ser realizada utilizando um filtro de primeira ordem, como na Equação 2.1:

$$y(t) = x(t) - \alpha * x(t-1) \tag{2.1}$$

onde o valor de α é comumente adotado entre 0.95 e 0.97 (CARVALHO, 2020).

Essa abordagem é particularmente útil para compensar as perdas de alta frequência que podem ocorrer durante a gravação, transmissão ou armazenamento do áudio. Ao aumentar a ênfase nas frequências mais altas, a pré-enfase ajuda a melhorar a inteligibilidade e a percepção do áudio, resultando em uma experiência auditiva mais agradável para os ouvintes.

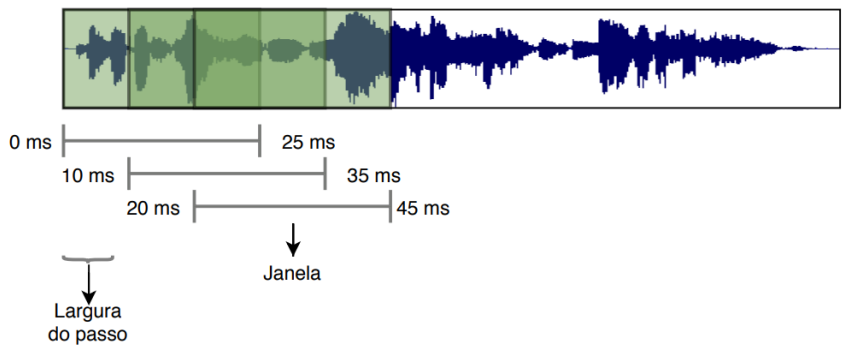
Essa técnica é amplamente utilizada em sistemas de comunicação, como telefonia, transmissões de rádio e televisão, bem como em aplicações de áudio em geral, como gravações musicais e produções de áudio.

2.3.2 Janela deslizante

A janela deslizante, também conhecida como janela móvel ou janela de deslocamento, é uma técnica amplamente utilizada em processamento de sinais e análise de dados sequen-

ciais. Essa abordagem consiste em dividir uma sequência de dados em pequenos segmentos, denominados janelas, e processar cada janela individualmente (CARVALHO, 2020).

Figura 1 – Exemplo de janela deslizante.



Fonte: CARVALHO (2020)

Na Figura 1, tem-se um exemplo de janela deslizante. Frequentemente, um valor adotado para a largura da janela é de 25 ms, enquanto a largura do passo é de 10 ms, resultando em uma sobreposição de 15 ms entre janelas consecutivas.

Para avançar na análise, a janela é deslocada em incrementos fixos ou sobrepostos ao longo da sequência de dados. Essa técnica é especialmente útil em aplicações como processamento de áudio, processamento de imagens, análise de séries temporais e processamento de linguagem natural (CAMASTRA; VINCIARELLI, 2015).

A janela deslizante permite que informações locais e padrões presentes em diferentes partes da sequência sejam capturados de forma eficiente, proporcionando uma visão mais detalhada e abrangente dos dados em análise. Além disso, é frequentemente empregada em algoritmos de extração de recursos, como no cálculo de espectrogramas para análise de frequências em sinais de áudio ou em algoritmos de processamento de texto que buscam padrões em documentos extensos (CAMASTRA; VINCIARELLI, 2015).

Usar a janela deslizante após a pré-ênfase é uma prática comum em processamento de sinais, especialmente em aplicações de análise espectral, como no caso de áudio ou processamento de fala (CARVALHO, 2020). A pré-ênfase é aplicada para realçar as altas frequências do sinal de áudio antes de sua transmissão ou armazenamento, a fim de melhorar a qualidade e a clareza do áudio. Essa ênfase nas altas frequências é útil porque muitas vezes essas frequências são mais suscetíveis a perdas durante a transmissão ou gravação do sinal.

Após a pré-ênfase, o uso da janela deslizante é benéfico para várias finalidades, incluindo:

1. Segmentação dos dados: A janela deslizante divide o sinal pré-enfatizado em segmentos menores, permitindo a análise localizada de pequenas partes do sinal de cada vez. Isso é particularmente útil quando se deseja analisar padrões ou eventos específicos em diferentes partes do sinal.

2. Redução do efeito de borda: A aplicação de janelas deslizantes com sobreposição pode ajudar a reduzir o efeito de borda indesejado que ocorre quando as bordas dos segmentos do sinal são abruptamente cortadas. A sobreposição suaviza a transição entre os segmentos, minimizando assim o impacto das bordas nos resultados da análise.

3. Análise espectral local: Ao aplicar a janela deslizante, pode-se calcular a transformada de Fourier em cada janela, resultando em uma representação espectral local do sinal. Essa abordagem é comumente usada em cálculos de espectrogramas, permitindo a visualização das variações de frequência ao longo do tempo.

4. Eficiência computacional: O uso de janelas deslizantes reduz o esforço computacional ao processar o sinal em partes menores, em vez de analisar todo o sinal de uma só vez.

Em resumo, a combinação da pré-ênfase e da janela deslizante é uma estratégia eficaz para a análise e processamento de sinais, permitindo extrair informações relevantes, melhorar a representação espectral e lidar com o efeito de borda de forma mais adequada (CAMASTRA; VINCIARELLI, 2015).

2.3.3 Janelamento

O janelamento é uma técnica crucial em processamento de sinais e análise espectral, usada para segmentar dados contínuos em trechos menores e mais manejáveis, denominados janelas. Essas janelas são geralmente definidas por funções específicas, chamadas de funções de janela, que atribuem pesos aos dados contidos em cada segmento.

A escolha adequada da função de janela pode influenciar significativamente os resultados da análise, uma vez que diferentes funções de janela têm propriedades distintas, como a forma e a largura do lóbulo principal, o nível de vazamento espectral e a capacidade de reduzir o efeito de borda.

As funções de janela mais comuns incluem a janela retangular, a janela de *Hamming*, a janela de *Hanning* e a janela de *Blackman*, cada uma delas adequada para diferentes situações e aplicações. Na Equação 2.2, está representado o janelamento de *Hamming*.

$$w(n) = 0.54 - 0.46 * cos\left(\frac{2\pi n}{N-1}\right) \tag{2.2}$$

O janelamento é amplamente empregado em cálculos espectrais, como na transformada de Fourier, no cálculo de espectrogramas, na análise de frequência de sinais de áudio e em muitos outros campos onde é necessário analisar dados sequenciais em segmentos menores para uma melhor compreensão e interpretação das informações contidas no sinal (CAMASTRA; VINCIARELLI, 2015).

2.3.4 Transformada de Fourier

A Transformada de Fourier após o janelamento é uma técnica essencial em análise espectral e processamento de sinais. O janelamento é aplicado para dividir um sinal contínuo em segmentos menores, denominados janelas, e a Transformada de Fourier é aplicada individualmente a cada janela. Essa abordagem permite obter uma representação espectral localizada do sinal em diferentes momentos ao longo do tempo, revelando como as frequências variam em cada janela.

A Transformada de Fourier após o janelamento é especialmente útil em situações em que o sinal não é estacionário, ou seja, suas características podem mudar ao longo do tempo. Nesses casos, a aplicação da Transformada de Fourier diretamente ao sinal inteiro poderia obscurecer as variações temporais das frequências. Ao contrário, a Transformada de Fourier após o janelamento fornece uma visão mais detalhada e dinâmica do espectro de frequências, permitindo a análise e a detecção de mudanças significativas no comportamento do sinal ao longo do tempo.

A Transformada de Fourier é uma poderosa ferramenta matemática utilizada para analisar sinais e descrever fenômenos no domínio da frequência. Ela permite decompor um sinal em suas componentes de frequência, revelando as diferentes frequências que compõem o sinal original (CAMASTRA; VINCIARELLI, 2015).

Ao aplicar a Transformada de Fourier em um sinal no domínio do tempo, obtém-se um espectro de frequências no domínio da frequência, que representa a contribuição de cada frequência presente no sinal. Essa transformação é especialmente útil em aplicações como processamento de áudio, análise de séries temporais, análise de sinais de comunicação, processamento de imagens, entre outras áreas em que a informação contida nas diferentes frequências e sua evolução temporal são fundamentais para a compreensão dos dados e a tomada de decisões (CAMASTRA; VINCIARELLI, 2015).

A Transformada de Fourier tem sido amplamente utilizada em uma variedade de campos científicos e tecnológicos, proporcionando *insights* valiosos sobre as características e comportamentos dos sinais, e desempenha um papel fundamental na engenharia de sinais e sistemas. Além disso, a Transformada de Fourier tem inúmeras extensões e variantes, como a Transformada de Fourier de curta duração (STFT), que é aplicada para analisar sinais não estacionários em pequenos intervalos de tempo, tornando-a ainda mais versátil e amplamente aplicável em diversas áreas de estudo e pesquisa.

A STFT é especialmente indicada para sinais que apresentam componentes com variações significativas ao longo do tempo, como é o caso da fala. Enquanto a transformada de Fourier clássica calcula o espectro de frequências considerando todo o sinal como um todo, a STFT atua em regiões localizadas, fornecendo informações de frequência mais precisas e permitindo distinguir as variações das frequências do sinal ao longo do tempo. Isso torna a STFT uma ferramenta valiosa para analisar e entender melhor as características temporais e espectrais de sinais com comportamentos dinâmicos, como a fala (CARVALHO, 2020).

A STFT é dada pelas Equações 2.3 e 2.4:

$$X_{STFT}(m,n) = \sum_{k=0}^{L-1} x(k) \times g(k-m) \times e^{\frac{-2j\pi nk}{L}} \tag{2.3}$$

$$X(k) = \sum_{m}\sum_{n} X_{STFT}(m,n) \times g(k-m) \times e^{\frac{-2j\pi nk}{L}} \tag{2.4}$$

onde $x(k)$ representa o sinal e $g(k)$ representa um dos L janelamentos. A partir das Equações 2.3 e 2.4, a STFT de $x(k)$ pode ser interpretada como a transformada de Fourier do produto de $x(k) \times g(k-m)$. A $X_{STFT}(m,n)$ é o resultado da STFT de uma janela e $X(k)$ é a STFT completa do sinal de fala.

2.3.5 Espectro de Potência

O espectro de potência é uma importante representação de sinais de áudio que fornece informações valiosas sobre as características de frequência presentes em uma gravação de som. Em termos simples, o espectro de potência mostra como a energia do sinal de áudio está distribuída ao longo das diferentes frequências.

O espectro de potência é obtido através de uma análise espectral, que geralmente é realizada usando a STFT. Uma vez que a análise espectral é realizada, o espectro de potência é representado graficamente em um gráfico, onde o eixo horizontal representa as diferentes frequências e o eixo vertical mostra a magnitude da energia em cada frequência. Áreas com maior magnitude indicam a presença de frequências mais proeminentes no sinal de áudio.

A partir do resultado da etapa anterior, procede-se ao cálculo da potência espectral do sinal utilizando a Equação 2.5:

$$S(k) = |X(k)|^2 = [X(k)]^2 + [jX(k)]^2 \tag{2.5}$$

O propósito desse procedimento é amplificar as quantidades de energia em diversas faixas de frequência. Essa etapa é crucial, pois leva em conta as altas frequências do sinal, que podem conter informações relevantes para melhorar o desempenho dos classificadores.

Um exemplo significativo são os sons vocálicos, os quais frequentemente possuem uma maior concentração de energia em áreas de baixa frequência. No entanto, as informações contidas nas faixas de frequências mais altas são frequentemente de extrema importância para distinguir algumas vogais, que possuem pronúncias semelhantes (CARVALHO, 2020).

O espectro de potência é amplamente utilizado em diversas aplicações de processamento de áudio, incluindo equalização de áudio, remoção de ruído, análise de acústica de salas, síntese sonora e muito mais. É uma ferramenta essencial para profissionais de áudio, músicos, engenheiros de som e cientistas que desejam compreender e manipular o conteúdo espectral de sinais de áudio para diversos fins criativos e técnicos.

2.3.6 Escala Mel

O procedimento de converter o espectro de potência para a escala Mel é um método que modela a resposta em frequência de forma análoga ao sistema auditivo humano (LIU *et al.*, 2017). Essa técnica busca representar as características perceptivas da audição humana, que não percebe as frequências de maneira linear, mas sim de acordo com uma escala específica, conhecida como escala Mel.

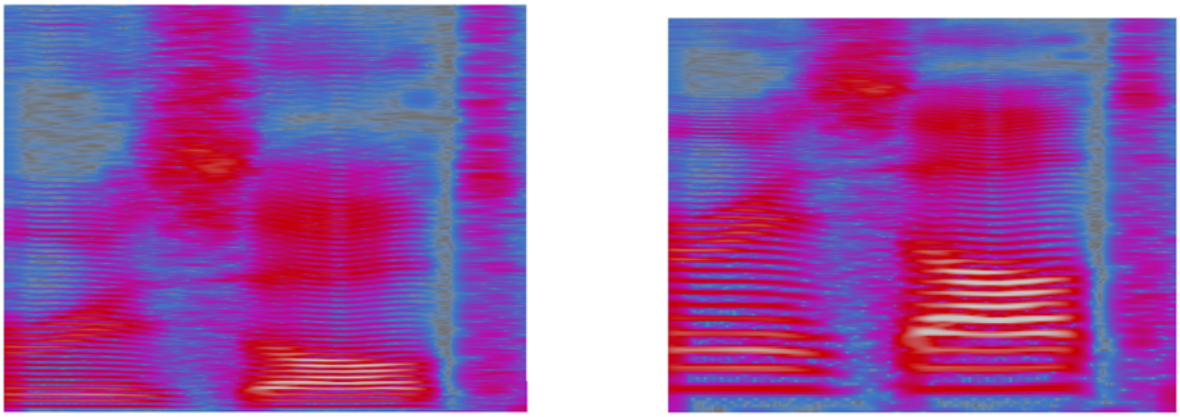
Dessa forma, a transformação para a escala Mel permite uma melhor aproximação das sensações auditivas humanas, tornando-a mais adequada para muitas aplicações em processamento de áudio, especialmente aquelas relacionadas à percepção sonora e ao reconhecimento de padrões auditivos.

A fim de realizar a transformação do espectro de potência para a escala Mel, é necessário aplicar um banco de K filtros à potência espectral $S[k]$. Esse banco de filtros consiste em filtros espaçados de acordo com a escala de frequência Mel, que é representada pela Equação 2.6:

$$Mel(f) = 1125 \times ln\left(1 + \frac{f}{700}\right) \tag{2.6}$$

Na prática, ao utilizar o espectrograma na escala Mel, ocorre um espaçamento entre as frequências de baixa magnitude localizadas na parte inferior do espectrograma. Ao mesmo tempo, as frequências de alta magnitude são ligeiramente compactadas, conforme a Figura 2:

Figura 2 – Comparação ilustrando o efeito da aplicação da escala Mel ao sinal.



(a) Exibição do espectro na escala linear.

(b) Exibição do espectro na escala Mel.

Fonte: CARVALHO (2020)

3 REDES *DEEP FEEDFORWARD*

Este capítulo aborda as redes *deep stacked autoencoders*. A Seção 3.1 apresenta o conceito de redes neurais artificiais, sua estrutura e aprendizagem. Na Seção 3.2, é apresentado o conceito de *autoencoder*, o seu objetivo de modo geral e os tipos de redes *deep autoencoders*.

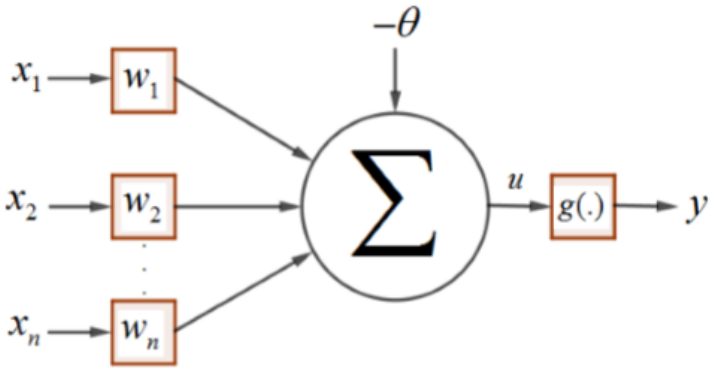
3.1 REDES NEURAIS ARTIFICIAIS

Uma rede neural artificial (RNA) consiste em unidades de processamento densamente interconectadas, chamadas neurônios artificiais, porque se comportam de forma semelhante aos neurônios biológicos. O sistema nervoso é formado por um conjunto extremamente complexo de células (neurônios biológicos), responsáveis pelo funcionamento e comportamento do corpo humano e do raciocínio.

Em 1943 o médico psiquiatra e neuroanatomista Warren S. McCulloch, juntamente ao matemático Walter Pitts descreveram o primeiro modelo artificial que foi uma simplificação do que era conhecido na época de neurônios biológicos (AZEVEDO, 2016). Esse conceito foi trabalhado por Rosenblatt que desenvolveu as RNAs baseadas no *perceptron*. Segundo Nascimento (2015), de uma maneira simplificada, uma RNA é um modelo computacional/matemático inspirado no sistema nervoso de seres vivos e que adquire conhecimento através da experiência. Os neurônios artificiais são interligados por um enorme número de interconexões (sinapses artificiais) representadas por vetores/matrizes de peso sináptico.

O neurônio artificial assemelha-se a um neurônio biológico. Desse modo, os componentes são cambiados da seguinte maneira: dendritos pelas entradas, as ligações com o corpo celular são feitas através de pesos (assemelhando as sinapses), os estímulos identificados pelos dendritos (entradas) são processados pela função de soma e o axônio é substituído pela função de ativação. Na Figura 3, está a representação de um neurônio artificial.

Figura 3 – Neurônio artificial.



Fonte: (CAMPOS, 2018)

A partir da observação do processo biológico, foram encontrados os sete elementos base análogos com o sistema artificial, que compõe o neurônio de acordo com os conceitos de Silva e Santos (2015 apud CAMPOS, 2018):

- Sinais de entrada $[X_1, X_2, X_3 ... X_N]$: são sinais ou medidas provenientes do meio externo, representando valores assumidos pelas variáveis de uma aplicação intrínseca;
- Pesos sinápticos $[W_1, W_2, W_3 ... W_N]$: são os valores que ponderam cada uma das variáveis de entrada;
- Combinador linear $[\sum]$: possui a função de agregar todos os sinais de entrada, ponderados pelos respectivos pesos sinápticos, com o objetivo de produzir um valor de potencial de ativação;
- Limiar de ativação (bias) $[\theta]$: é uma variável que caracteriza o patamar apropriado para que o resultado determinado pelo combinador linear possa gerar um valor de disparo em direção a saída do neurônio;
- Potencial de ativação: é o resultado produzido pela diferença entre o valor gerado pelo combinador linear e o limiar de ativação;
- Função de ativação $[g]$: seu objetivo é limitar a saída do neurônio dentro de um intervalo de valores razoáveis, assumidos pela sua própria imagem funcional;
- Sinal de saída $[y]$: é o valor final produzido pelo neurônio em relação a um determinado conjunto de sinais de entrada.

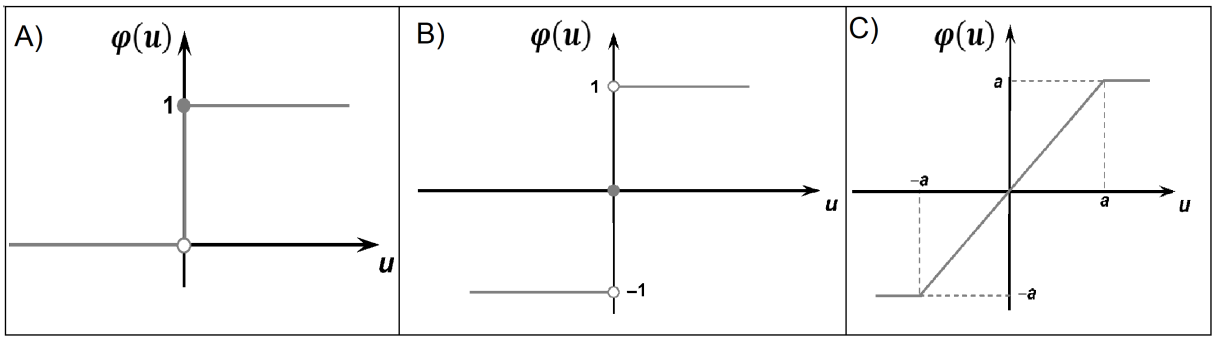
O resultado gerado pelo neurônio artificial é fruto das funções abaixo, que foram propostas por McCullon e Pitts. A representação matemática do neurônio artificial está na Equação 3.1.

$$y = g\left(\sum_{i=1}^{n} w_i \times x_i - \theta\right) \tag{3.1}$$

Com base na eficiência dos processos realizados pelo cérebro e inspirados pelo seu funcionamento, vários pesquisadores têm desenvolvido há mais de 30 anos a teoria das RNAs, as aprender estratégias de solução baseadas em exemplos de comportamento típico de padrões (HAYKIN, 2001).

As funções de ativação parcialmente diferenciáveis podem ser conceituadas como: função degrau, degrau bipolar ou sinal e rampa simétrica; e possui as funções totalmente diferenciáveis que são: função sigmoide, tangente hiperbólica, gaussiana e linear (CAMPOS, 2018). Na Figura 4, estão as funções de ativação parcialmente diferenciáveis.

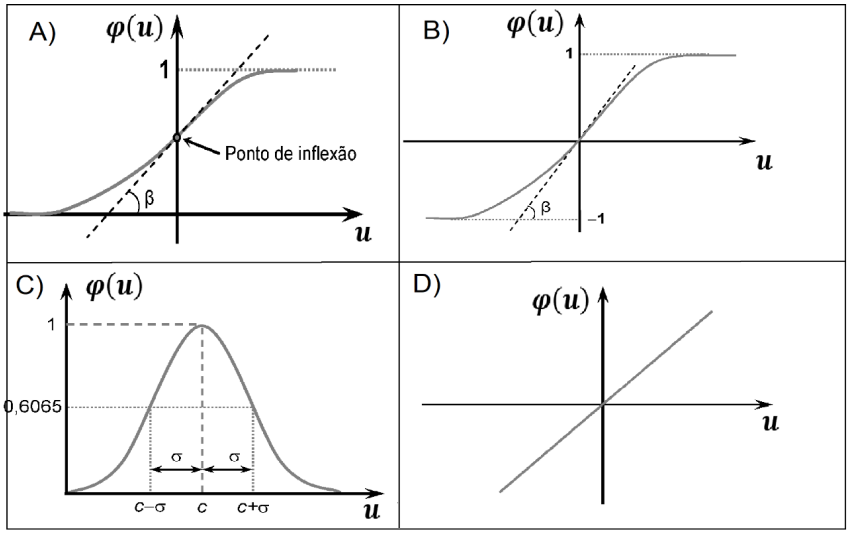
Figura 4 – Funções de ativação parcialmente diferenciáveis: A) Função Degrau, B) Função Degrau Bipolar e C) Função Rampa Simétrica.



Fonte: Flauzino *et al.* (2010)

Na Figura 5, estão as funções de ativação totalmente diferenciáveis.

Figura 5 – Gráficos das Funções Totalmente Diferenciáveis: A) Função logística, B) Função tangente hiperbólica, C) Função Gaussiana e D) Função Linear.



Fonte: Flauzino *et al.* (2010)

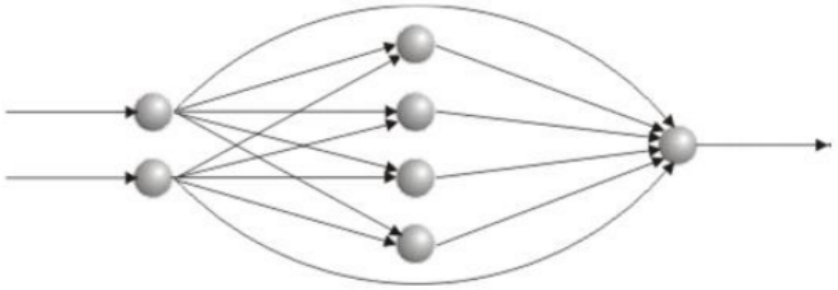
A função de ativação limita o valor de saída do neurônio, normalmente está no intervalo entre 0 e 1 ou -1 e 1, sendo utilizadas como uma função de transferência de valores entre neurônios (AZEVEDO, 2016, p.21).

A estrutura de uma RNA é segmentada por camadas, sendo a primeira camada conhecida como camada de entrada, a segunda é chamada de intermediária, oculta, escondida ou invisível e a última camada, camada de saída. Entre as camadas, existem conexões que permitem fluxo de dados de uma camada para outra. O arranjo dessas conexões determina a arquitetura da rede. As arquiteturas mais utilizadas das RNAs são: *feedforward* e recorrentes.

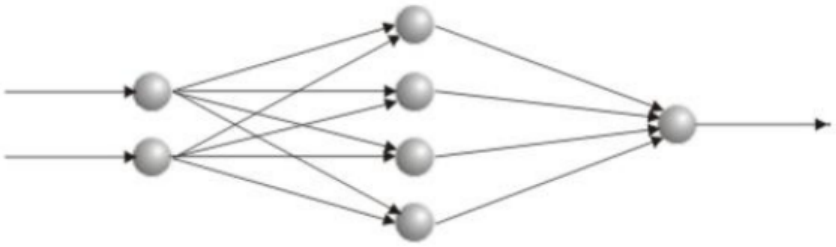
No modelo *feedforward*, o arranjo da RNA é constituído de uma camada de entrada, uma ou mais camadas intermediárias e uma camada de saída, não havendo interações entre neurônios da mesma camada. O fluxo de dados ocorre somente em uma direção, sempre o sentido da

entrada para a saída, ou seja, sem realimentação de dados (GOODFELLOW *et al.*, 2016; VIANA, 2019). Na Figura 6, mostram-se dois exemplos de RNAs *feedforward*.

Figura 6 – RNAs *feedfoward*.



(a) Topologia *feedforward*.



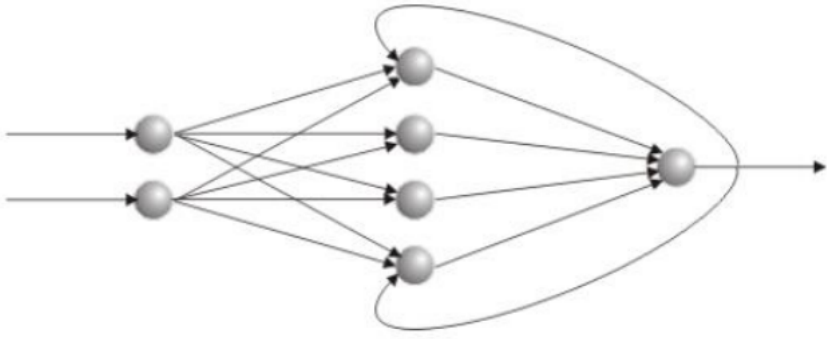
(b) Topologia estritamente *feedforward*.

Fonte: Almeida Neto (2003 apud VIANA, 2019)

Com esse tipo de rede, pode-se resolver casos como aproximação de funções, identificação de sistemas, reconhecimento de padrões, otimização, entre outros (CAMPOS, 2018).

Já as redes recorrentes, também chamadas de *feedbacks*, apresentam laço de recorrência, ou seja, existe ligação de neurônios de uma camada para neurônios da mesma camada ou de camadas anteriores (GOODFELLOW *et al.*, 2016; VIANA, 2019). Esse modelo de estrutura pode ser empregado em previsões de séries temporais, controle de processos, entre outras (CAMPOS, 2018, p.27). Na Figura 7, apresenta-se a estrutura de uma RNA recorrente.

Figura 7 – RNA *feedback*.



Fonte: Almeida Neto (2003 apud VIANA, 2019)

Existem diversos tipos de redes neurais artificiais, cada uma com suas próprias características e aplicações específicas. As redes neurais chamadas de Perceptron de múltiplas camadas (MLP) são amplamente utilizadas em atividades científicas e constituem o tipo de rede empregada neste trabalho. Derivadas da rede Perceptron de única camada, as MLPs possuem uma estrutura composta por no mínimo três camadas: a camada de entrada, uma camada intermediária e a camada de saída.

A quantidade de camadas escondidas e de neurônios na rede MLP não é calculada, mas sim estimada e ajustada com base na experiência do projetista da rede. No entanto, é necessário ter cuidado, pois um número excessivo de camadas intermediárias pode comprometer a convergência global, uma vez que os algoritmos de treinamento utilizados se baseiam no cálculo do erro da rede para ajustar os pesos.

O processo de aprendizado envolve o treinamento da rede utilizando um conjunto de amostras contendo dados de entrada ou dados de entrada e de saída. O objetivo é capacitar a rede para que ela responda corretamente na sua saída quando uma entrada diferente das utilizadas no treinamento for apresentada. Esse tipo de aprendizado pode ser realizado de forma supervisionada, não supervisionada ou semi supervisionada, dependendo da abordagem adotada (HAYKIN, 2001).

Na implementação desses tipos de aprendizado, é selecionado um algoritmo de treinamento que fornece as regras para ajuste dos pesos. Existem vários algoritmos disponíveis para cada tipo de RNA e constantemente novos modelos ou variações são apresentados pela comunidade científica. Neste trabalho, foi utilizado o algoritmo Backpropagation para o treinamento da rede CollabNet.

O algoritmo Backpropagation foi apresentado em 1986 e trouxe uma solução para o desafio de treinar redes neurais com várias camadas. Até então, não havia nenhum algoritmo capaz de atualizar todos os pesos da rede com base no erro de saída. O Backpropagation implementa o aprendizado do tipo supervisionado, visando a minimizar o erro quadrático médio de saída da RNA. Para isso, utiliza o método do gradiente descendente (VIANA, 2019).

O algoritmo Backpropagation funciona em duas fases: a fase *forward* e a fase *backward*. Na primeira fase, não há alteração nos valores dos pesos. Os sinais de entrada são propagados até a camada de saída para calcular o erro da rede. Essa fase começa com a alimentação dos neurônios na camada de entrada, que simplesmente recebe os padrões de entrada para o processamento pela rede. O primeiro processamento real ocorre na primeira camada escondida. Por esse motivo, alguns autores não consideram a camada de entrada como uma camada de neurônios.

3.2 REDES *DEEP AUTOENCODERS*

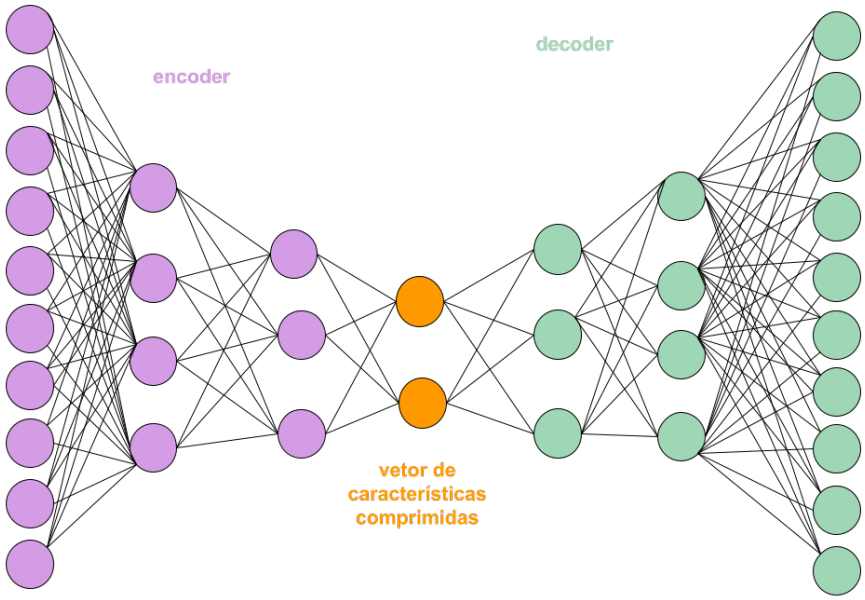
As DNNs são MLPs com muitas camadas ocultas, cujos pesos estão totalmente conectados e são muitas vezes (embora nem sempre) inicializados usando uma técnica de pré-treinamento não supervisionada ou supervisionada. As redes *deep autoencoders* é um tipo especial de DNN ,

porém sem etiquetas de classe, cujos vetores de saída têm a mesma dimensionalidade que os vetores de entrada. É frequentemente usada para aprender uma representação ou codificação eficaz dos dados originais, na forma de vetores de entrada, em camadas ocultas (DENG; YU, 2014).

Uma *autoencoder* é uma rede neural que é treinada para tentar copiar suas entradas ou saídas. Internamente, ela tem uma camada h oculta que descreve um código usado para representar a entrada, conforme a Figura 8. Uma rede neural *Autoencoder* é um algoritmo de aprendizado não supervisionado que aplica *Backpropagation*, definindo os valores de destino como iguais às entradas. Ou seja, usa $y_{(i)} = x_{(i)}$ (GOODFELLOW *et al.*, 2016).

A rede pode ser vista como consistindo de duas partes: uma função codificadora (*encoder*) $h = f(x)$ e um decodificador (*decoder*) que produz uma reconstrução $r = g(h)$ (GOODFELLOW *et al.*, 2016). Uma *deep autoencoder* é composta por duas redes *deep belief* simétricas. As camadas da *deep autoencoder* são máquinas de Boltzmann restritas (RBMs), os blocos de construção das *deep belief networks* (DATA SCIENCE ACADEMY, 2021).

Figura 8 – Rede *deep autoencoder*.



Fonte: Elaborado pela autora (2021).

A rede *deep autoencoder* é um método não-linear de extração de características sem utilizar etiquetas de classe. Como tal, as características extraídas visam conservar e representar melhor a informação em vez de realizar tarefas de classificação, embora às vezes estes dois objetivos estejam correlacionados. Quando treinada com um critério de *denoising*, uma *deep autoencoder* é também um modelo generativo e pode ser amostrada a partir dele (DENG; YU, 2014).

As *autoencoders* são projetadas para serem incapazes de aprender a copiar perfeitamente.

Normalmente elas são restritas de forma a permitir que copiem apenas aproximadamente e somente a entrada que se assemelha aos dados de treinamento. Como o modelo é forçado a priorizar quais aspectos da entrada devem ser copiados, muitas vezes ele aprende as propriedades úteis dos dados (GOODFELLOW *et al.*, 2016).

Uma *autoencoder* normalmente tem uma camada de entrada que representa os dados originais ou vetores da característica de entrada (por exemplo, *pixels* na imagem ou espectros na fala), uma ou mais camadas ocultas que representam a característica transformada, e uma camada de saída que combina com a camada de entrada para reconstrução. Quando o número de camadas ocultas é maior que um, a *autoencoder* é considerada profunda. A dimensão das camadas ocultas pode ser menor (quando o objetivo é a compressão da característica) ou maior (quando o objetivo é mapear a característica para um espaço de maior dimensão) do que a dimensão de entrada (DENG; YU, 2014).

A idéia das *autoencoders* tem sido parte da paisagem histórica das redes neurais por décadas (LECUN, 1987 ;BOURLARD e KAMP, 1988; HINTON e ZEMEL, 1994 apud Data Science Academy, 2021). Tradicionalmente, as *autoencoders* eram usadas para a redução da dimensionalidade ou aprendizagem de características. Recentemente, as conexões teóricas entre *autoencoders* e modelos variáveis latentes trouxeram as *autoencoders* à vanguarda da modelagem generativa.

As *autoencoders* podem ser consideradas como um caso especial de redes de alimentação e podem ser treinadas com todas as mesmas técnicas, tipicamente descendentes de gradientes de mini *batch* que seguem gradientes computados pelo *Backpropagation*. Ao contrário das redes de alimentação geral, as *autoencoders* também podem ser treinadas usando a circulação (HINTON e MCCLELLAND, 1988 apud Data Science Academy, 2021), um algoritmo de aprendizado baseado na comparação das ativações da rede com a entrada original com as ativações na entrada reconstruída. A recirculação é considerada mais plausível biologicamente do que o *backpropagation*, mas é raramente usada para aplicações de aprendizagem de máquinas.

As *autoencoders* são uma forma muito proveitosa de reduzir a dimensionalidade não linear devido a alguns motivos: elas fornecem mapeamentos flexíveis em ambos os sentidos. O tempo de aprendizagem é linear (ou melhor) no número de casos de treinamento. E o modelo de codificação final é bastante compacto e rápido. Todavia, pode ser muito difícil otimizar as *autoencoders* usando *backpropagation*. Com pequenos pesos iniciais, o gradiente do *backpropagation* morre. Mas existem outras formas de otimizá-las, usando o pré-treinamento camada-por-camada sem supervisão ou apenas inicializando os pesos com cuidado.

Uma RNA é frequentemente treinada usando uma das muitas variações do algoritmo *Backpropagation*, tipicamente o método de descida de gradiente estocástico. Embora muitas vezes razoavelmente eficaz, existem problemas fundamentais quando se utiliza a retropropagação do erro para treinar redes com muitas camadas ocultas. Uma vez que os erros são retropropagados para as primeiras camadas, eles se tornam minúsculos e o treinamento se torna bastante ineficaz. Por isso, uma rede perceptron multicamadas não pode ser uma rede profunda, ela se tornaria

ineficiente.

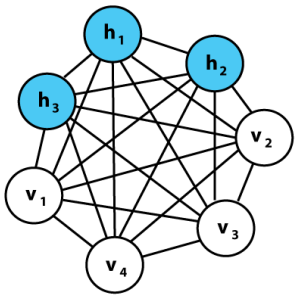
Embora métodos mais avançados de retropropagação ajudem em certa medida com este problema, ainda resulta em um aprendizado lento e em soluções ruins, especialmente com quantidades limitadas de dados de treinamento. Este problema pode ser aliviado pelo pré-treinamento de cada camada com uma simples *autoencoder* (BENGIO *et al.*, 2006). Esta estratégia tem sido aplicada em alguns trabalhos, onde foi construída uma *deep autoencoder* para mapear imagens em código binário curto para recuperação rápida de imagens, para codificar documentos (chamado hashing semântico) e para codificar características de fala em forma de espectrograma. O trabalho de Deng *et al.* (2010) é um exemplo do uso de uma *deep autoencoder* para extrair características de fala.

3.2.1 *Boltzmann machine*

Antes de abordar os tipos de redes *deep autoencoders*, é importante versar sobre a *Boltzmann machine* (na língua portuguesa, máquina de Boltzmann - BM). É uma rede de ligação simétrica. Suas unidades são semelhantes a neurônios que tomam decisões estocásticas sobre se devem estar ligadas ou não. Embora tenham um algoritmo de aprendizagem simples, as BMs gerais são muito complexas de estudar e muito lentas de treinar (DENG; YU, 2014).

As máquinas de Boltzmann são redes neurais estocásticas e generativas capazes de aprender representações internas e são capazes de representar e (com tempo suficiente) resolver difíceis problemas combinatórios. Têm o nome da distribuição Boltzmann (também conhecida como distribuição Gibbs), que é parte integrante da Mecânica Estatística e facilita a compreensão do impacto de parâmetros como a entropia e a temperatura nos estados quânticos na Termodinâmica. É por isso que são chamados de modelos baseados em energia (EBM). Foram inventados em 1985 por Geoffrey Hinton e Terry Sejnowski (SHARMA, 2018). Pode-se observar um exemplo de BM na Figura 9.

Figura 9 – Exemplo de *Boltzmann machine*.



Fonte: Sunny vd on Wikimedia

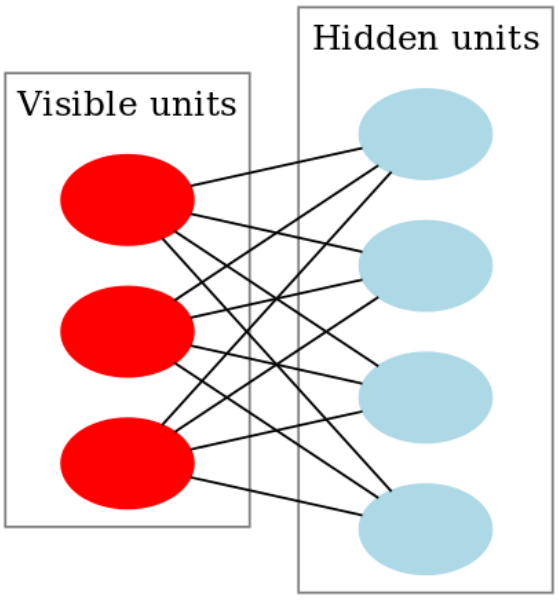
As máquinas de Boltzmann são modelos não determinísticos (ou estocásticos) generativos de aprendizagem profunda com apenas dois tipos de nós - nós ocultos e visíveis. Não há nós de saída e isto é que lhes dá esta característica não determinística. Não têm o tipo típico de saída 1

ou 0 através do qual os padrões são aprendidos e otimizados usando o gradiente descendente estocástico. Elas aprendem padrões sem essa capacidade. Pode-se ver pela Figura 9 que todos os nós estão ligados a todos os outros nós, independentemente de serem nós de entrada ou nós ocultos. Isto permite-lhes partilhar informação entre si e auto-gerar dados subsequentes. É mensurado apenas o que está nos nós visíveis e não o que está nos nós ocultos. Quando a entrada é fornecida, eles são capazes de capturar todos os parâmetros, padrões e correlações entre os dados. É por isso que são chamados modelos generativos profundos e se enquadram na classe de aprendizagem profunda não supervisionada (SHARMA, 2018).

3.2.1.1 Restricted Boltzmann machine

A *restricted Boltzmann machine* (na lingua portuguesa, máquina de Boltzmann restrita - RBM) é um tipo especial de BM que consiste de uma camada de unidades visíveis e uma camada de unidades ocultas sem conexões visíveis ou ocultas. São redes neurais artificiais de duas camadas, conforme a Figura 10, com capacidades generativas. Têm a capacidade de aprender uma distribuição de probabilidade sobre o seu conjunto de input. As RBMs foram inventadas por Geoffrey Hinton e podem ser utilizados para redução de dimensionalidade, classificação, regressão, filtragem colaborativa, aprendizagem de características e modelação de tópicos (DENG; YU, 2014).

Figura 10 – Exemplo de *Restricted Boltzmann machine*.



Fonte: Sharma (2018)

As RBMs são uma classe especial de máquinas Boltzmann e são restritos em termos de ligações entre as unidades visíveis e as unidades ocultas. Isto facilita a sua implementação quando comparadas com as máquinas de Boltzmann. Como foi dito anteriormente, são uma rede neural de duas camadas (sendo uma a camada visível e a outra a camada oculta) e estas

duas camadas estão ligadas por um gráfico totalmente bipartido. Isto significa que cada nó da camada visível está ligado a cada nó da camada oculta, mas não há dois nós no mesmo grupo que estejam ligados um ao outro. Esta restrição permite algoritmos de treino mais eficientes do que os disponíveis para a classe geral das máquinas de Boltzmann, em particular, o algoritmo de divergência contrastiva baseada no gradiente (SHARMA, 2018).

Em um RBM, conforme a Figura 10, tem-se um gráfico bipartido simétrico onde não estão ligadas duas unidades dentro do mesmo grupo. Múltiplos RBMs também podem ser empilhados e podem ser afinados através do processo de descida por gradiente e retropropagação. Tal rede é chamada rede de crença profunda. A RBM é uma rede neural estocástica, o que significa que cada neurônio terá algum comportamento aleatório quando ativado. Existem duas outras camadas de unidades de polarização (polarização oculta e polarização visível) num RBM. Isto é o que torna as RBMs diferentes dos *autoencoders*. A polarização oculta da RBM produz a ativação no passe para a frente e a polarização visível ajuda a RBM a reconstruir a entrada durante um passe para trás. A entrada reconstruída é sempre diferente da entrada real, uma vez que não existem ligações entre as unidades visíveis e, portanto, nenhuma forma de transferência de informação entre elas (SHARMA, 2018).

3.3 TIPOS/ARQUITETURAS DE *DEEP AUTOENCODERS*

Nesta seção, foram apresentados as redes: *stacked autoencoders* tradicionais, *deep belief networks*, *deep Boltzmann machines* e *stacked denoising autoencoders*. Apresentando de cada uma, sua definição, características, métodos de aprendizado, vantagens, desvantagens e aplicações.

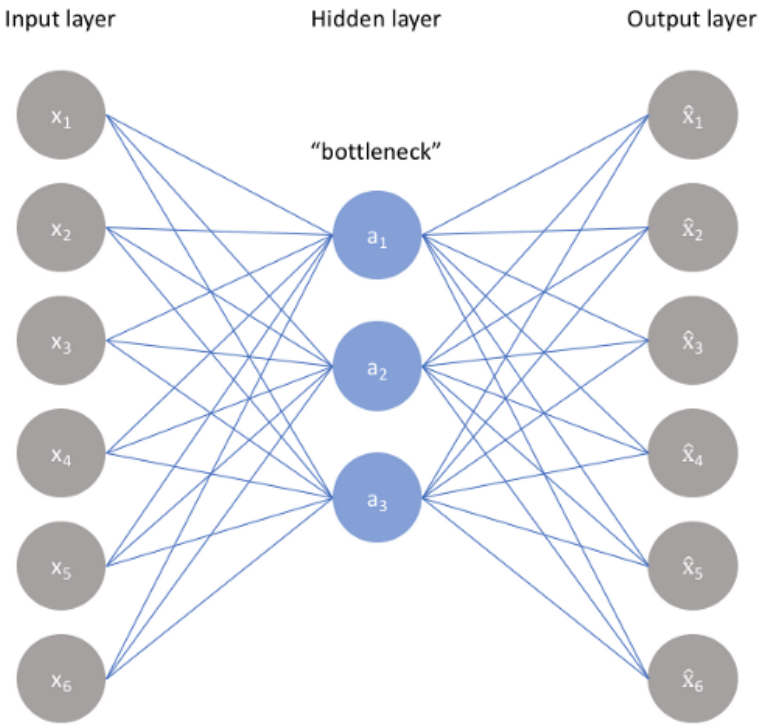
3.3.1 *Stacked autoencoder* tradicional

A *stacked autoencoder* (DSN) é um rede neural composta por diversas camadas de sparse autoencoders, em que, as entradas de cada camada é ligada às saídas da camada posteriores (CHAUDHARY, 2019). Ela foi originalmente concebida tendo em mente o problema da escalabilidade da aprendizagem (DENG; YU, 2014). A arquitetura básica do *autoencoder* é composta de três camadas: duas externas, nomeadas de *encoder* e *decoder*, e uma interna ou oculta, a *bottleneck*. O objetivo dessa rede neural é que a saída seja igual a entrada, em um processo de codificação da camada de entrada para a oculta e decodificação da camada oculta para saída, e durante esse processo a *bottleneck* obtém uma função relacionada aos dados resultantes (LIU *et al.*, 2017).

Na Figura 11, é possível notar mais camadas ocultas do que camadas externas, o que está relacionado a técnica de forçar uma extração real de recursos de entradas, nesse caso escolhendo as principais características ou diminuindo a dimensão dos dados (LIU *et al.*, 2017).

A ideia central da DSN se relaciona com o conceito de empilhamento, tal como proposto e explorado em Bengio *et al.* (2006), onde módulos simples de funções ou classificadores são

Figura 11 – Arquitetura *autoencoder*.



Fonte: Elaborado pela autora (2023).

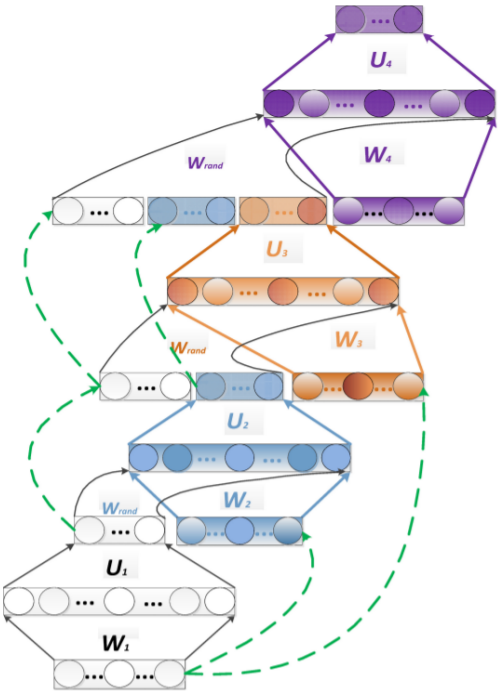
compostos primeiro e depois são "empilhados"uns sobre os outros, a fim de aprender funções ou classificadores complexos. A arquitetura DSN foi originalmente apresentada em Deng e Yu (2011) e foi referida como rede convexa profunda para enfatizar a natureza convexa de uma parte importante do algoritmo utilizado para a aprendizagem da rede.

O nome “convexo” utilizado em Deng e Yu (2011) acentua o papel da otimização convexa na aprendizagem dos pesos da rede de saída, dadas as atividades das unidades ocultas em cada módulo básico. Também aponta para a importância das restrições de forma fechada, derivadas da convexidade, entre os pesos de entrada e de saída. Tais restrições tornam a aprendizagem dos restantes parâmetros da rede (ou seja, os pesos da rede de entrada) muito mais fácil do que de outra forma, permitindo a aprendizagem em modo batch do DSN que pode ser distribuído por clusters de CPU. E em publicações mais recentes, o DSN foi utilizado quando a operação chave de empilhamento é enfatizada (DENG; YU, 2014).

A DSN faz uso de informação de supervisão para empilhar cada um dos módulos básicos, que assume a forma simplificada de perceptron de várias camadas. No módulo básico, as unidades de saída são lineares e as unidades ocultas são sigmoidais não lineares. A linearidade nas unidades de saída permite uma estimativa de forma altamente eficiente, paralela e fechada (um resultado de otimização convexa) para os pesos da rede de saída, dadas as atividades das unidades ocultas. Devido às restrições de forma fechada entre os pesos de entrada e saída, os pesos de entrada podem também ser elegantemente estimados de uma forma eficiente, paralela e em lote (DENG; YU, 2014).

Uma DSN, como mostrado na Figura 12, inclui um número variável de módulos em camadas, em que cada módulo é uma rede neural especializada composta por uma única camada oculta e dois conjuntos de pesos treináveis. Na Figura 12, apenas quatro desses módulos são ilustrados, em que cada módulo é mostrado com uma cor separada. Na prática, até algumas centenas de módulos foram eficientemente treinados e utilizados em experiências de classificação de imagem e fala (DENG; YU, 2014).

Figura 12 – Exemplo de *stacked autoencoder*.



Fonte: Deng e Yu (2014)

O módulo mais baixo na DSN compreende uma camada linear com um conjunto de unidades de entrada linear, uma camada não linear oculta com um conjunto de unidades não lineares, e uma segunda camada linear com um conjunto de unidades de saída linear. Uma não linearidade sigmoidal é tipicamente utilizada na camada oculta. No entanto, outras não linearidades também podem ser utilizadas. As unidades de saída na camada de saída linear representam os alvos da classificação.

A matriz de peso mais baixo, denotada por W, liga a camada de entrada linear e a camada não linear oculta. A camada superior a matriz de peso, denotada por U, liga a camada oculta não linear com a camada de saída linear. A matriz de peso U pode ser determinada através de uma solução de forma fechada dada a matriz de peso W quando é utilizado o critério de formação de erro quadrático médio.

Como indicado acima, a DSN inclui um conjunto de ligações em série, módulos sobrepostos e estratificados, em que cada módulo tem a mesmo arquitetura - uma camada de entrada linear seguida por uma camada não linear oculta camada, que está ligada a uma camada de saída linear. Nota-se que as unidades de saída de um módulo inferior são um subconjunto das

unidades de entrada de um módulo superior adjacente na DSN. Mais especificamente, num segundo módulo diretamente acima do módulo mais baixo na DSN, as unidades de entrada podem incluir as unidades de saída do módulo mais baixo e, opcionalmente, a característica de entrada bruta.

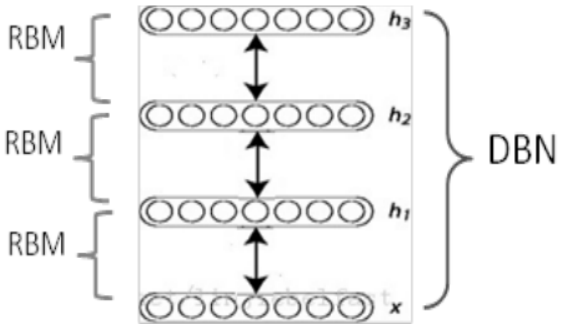
Esse padrão de incluir unidades de saída num módulo inferior como uma parte das unidades de entrada num módulo superior adjacente e depois aprende uma matriz de peso que descreve os pesos de ligação entre unidades ocultas e unidades de saída linear através da otimização convexa pode continuar para muitos módulos.

3.3.2 *Deep belief networks*

As *Deep belief networks* (DBNs) são modelos generativos probabilísticos compostos por várias camadas de variáveis ocultas estocásticas. Uma DBN é composta por uma pilha de máquinas de *Boltzmann* restritas (RBMs). Um componente central da DBN é um algoritmo de aprendizagem guloso, camada a camada, que otimiza os pesos da DBN com uma complexidade de tempo linear ao tamanho e à profundidade das redes. A inicialização dos pesos de uma MLP com uma DBN configurada de forma correspondente produz frequentemente resultados muito melhores do que com os pesos aleatórios (DENG; YU, 2014). Como tais, as MLPs com muitas camadas ocultas, ou redes neurais profundas (DNN), que são aprendidas com pré-treinamento não supervisionado da DBN seguido de ajuste via Backpropagation, são por vezes também denominadas de DBN na literatura. Mais recentemente, os pesquisadores têm sido mais cuidadosos na distinção entre DNNs e DBNs. Quando a DBN é usada para inicializar o treino de uma DNN, a rede resultante é por vezes é chamada DBN-DNN (DENG; YU, 2014).

Historicamente, a DBN foi construída com base em um novo pensamento sobre a arquitetura das RBMs, publicado em 2006 por Hinton *et al.* (2006). Então, quando se tem uma saída da camada oculta em uma RBM, pode-se usá-la como a entrada da camada visível de outra RBM. Esse processo pode ser considerado como uma extração adicional do recurso extraído das amostras, o que dessa forma, define uma DBN. Na Figura 13, tem-se um conjunto de camadas RBM, em que, a saída da camada superior é a entrada da camada inferior.

Figura 13 – Exemplo de DBN baseada em uma RBM.



Fonte: Hua *et al.* (2015)

Como propriedades atraentes da DBN, tem-se os bons pontos de inicialização do sistema de rede neural profunda. Outro ponto, o algoritmo de aprendizagem faz uso efetivo de dados não rotulados. Ainda, pode ser interpretado como um modelo generativo probabilístico. E por fim, o problema de superajuste, que é frequentemente observado nos modelos com milhões de parâmetros, como DBNs, e o problema de subajuste, que ocorre com frequência em redes profundas, pode ser efetivamente aliviado pela etapa de pré-treinamento ttgeradora (DENG; YU, 2014).

3.3.3 Deep boltzmann machines

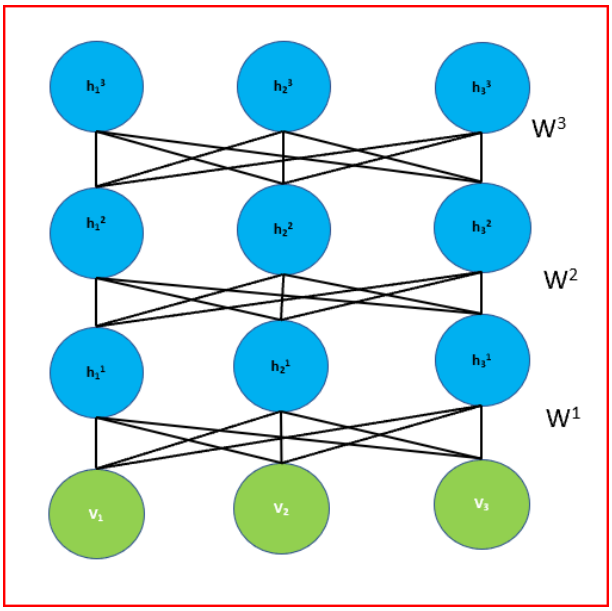
Outro tipo proeminente de modelos profundos não supervisionados com capacidade generativa é a *deep boltzmann machines* (na língua portuguesa, máquina de Boltzmann profunda - DBM). Uma DBM contém muitas camadas de variáveis ocultas e não tem conexões entre as variáveis dentro da mesma camada. Em uma DBM, cada camada captura correlações complicadas e de maior ordem entre as atividades de características ocultas na camada abaixo (DENG; YU, 2014).

As DBM têm o potencial de aprender representações internas que se tornam cada vez mais complexas, altamente desejáveis para resolver problemas de reconhecimento de objetos e de fala. Além disso, as representações de alto nível podem ser construídas a partir de um grande suprimento de entradas sensoriais não rotuladas e dados rotulados muito limitados podem então ser usados para afinar apenas ligeiramente o modelo para uma tarefa específica em mãos. Finalmente, ao contrário das DBN, o procedimento de inferência aproximada, para além de uma passagem inicial de baixo para cima, pode incorporar *feedback* de cima para baixo, permitindo às máquinas de Boltzmann profundas propagarem melhor a incerteza e, por conseguinte, lidar de forma mais robusta com entradas ambíguas (SALAKHUTDINOV; HINTON, 2009).

É um modelo sem supervisão, probabilístico, generativo, com ligações totalmente não direcionadas entre diferentes camadas. Contém unidades visíveis e múltiplas camadas ocultas. Assim como o RBM, não existe conexão entre camadas na DBM. Existem conexões somente entre as unidades das camadas vizinhas. É uma rede de unidades binárias estocásticas simetricamente ligadas. A DBM pode ser organizada como gráfico bipartido com camadas ímpares de um lado e camadas pares de um lado. As unidades dentro das camadas são independentes umas das outras, mas dependem das camadas vizinhas. O aprendizado se torna eficiente por meio do pré-treinamento camada por camada. O pré-treinamento guloso de camada é ligeiramente diferente do que é feito na DBM. Depois de aprender as características binárias em cada camada, o DBM é ajustado por retropropagação (KHANDELWAL, 2018).

Quando o número de camadas ocultas de uma DBM é reduzido a um, a máquina de Boltzmann é restrita (RBM). Assim como a DBM, não há conexões de oculta para oculta e de visível para visível na RBM. A principal virtude da RBM é que, através da composição de muitos RBMs, muitas camadas ocultas podem ser aprendidas eficientemente usando as ativações

Figura 14 – Exemplo de DBM.



Fonte: Khandelwal (2018)

de características de uma RBM como os dados de treinamento para o próximo. Tal composição leva a uma *deep belief network* (DBN) (DENG; YU, 2014).

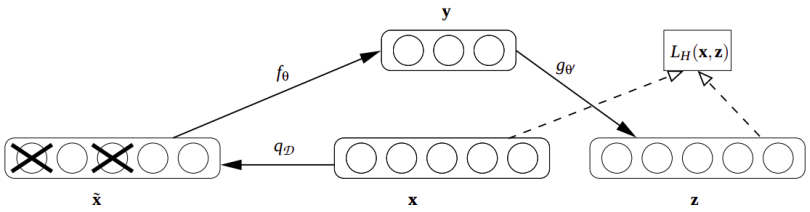
A máquina de Boltzmann utiliza cadeias de Markov inicializadas aleatoriamente para aproximar o gradiente da função de probabilidade, que é demasiado lento para ser prático. A DBM usa pré-treinamento guloso, de camada por camada, para acelerar o aprendizado dos pesos. Ela se baseia em pilhas de aprendizado da máquina de Boltzmann restrita com uma pequena modificação usando divergência contrastiva (KHANDELWAL, 2018).

A principal intuição para o treinamento guloso camada a camada para DBM é que a entrada é duplicada para o RBM de nível inferior e o RBM de nível superior. As entradas da RBM de nível inferior são duplicadas para compensar a falta de entrada de cima para baixo na primeira camada oculta. Da mesma forma para a RBM de nível superior, são duplicadas as unidades ocultas para compensar a falta de entrada de baixo para cima. Para as camadas intermediárias, os pesos da RBM são simplesmente duplicados (KHANDELWAL, 2018).

3.3.4 *Stacked denoising autoencoders*

A técnica de *denoising autoencoders* garante uma melhor extração das características úteis, evitando uma solução óbvia do processo de apenas copiar e colar a saída de uma camada superior para a entrada de uma camada inferior. Assim, ao invés de restringir a representação dos dados, removem-se os ruídos, como declaram Vincent *et al.* (2010). Na Figura 15, é exemplificado um processo de *denoising* para reconstrução de x a partir de uma representação corrompida e a qualidade do resultado é comparada entre a saída final z e o resultado esperado x (VINCENT *et al.*, 2010).

Figura 15 – Exemplo de Denoising.



Fonte: Vincent *et al.* (2010)

A *Stacked denoising autoencoders* para inicializar uma rede profunda funciona da mesma maneira que juntar RBMs em DBN ou *autoencoders* comuns. A corrupção dos dados de entrada é um mecanismo utilizado apenas para o treinamento de remoção do ruído inicial de cada camada individual, para que ele possa aprender a extrair os recursos úteis. Uma vez aprendido, ele é usado como entrada, para as camadas inferiores, não corrompidas. Dessa forma, nenhuma técnica de corrupção é aproveitada para gerar a representação que servirá como entrada limpa para o treinamento da próxima camada (VINCENT *et al.*, 2010).

4 METODOLOGIA

Este capítulo define a estrutura que foi utilizada neste trabalho. A Seção 4.1 apresenta as ferramentas utilizadas para o desenvolvimento do mesmo. Na Seção 4.2, apresenta-se os métodos propostos da pesquisa.

4.1 MATERIAIS

Para o pré-processamento de dados, foram usadas as linguagens Shell e Python para a construção de algoritmos, em conjunto com as ferramentas: o Praat, o UFPAlign e o Kaldi. As bases de dados usadas para estudo foram: a TIMIT Acoustic-Phonetic Continuous Speech Corpus, a CSLU: Spoltech Brazilian Portuguese Version 1.0, a LaPS Benchmark 16k e a Male/Female for Forced Phonetic Alignment, sendo as duas ultimas fornecidas pelo Grupo FalaBrasil da Universidade Federal do Pará, cuja a ultima foi aplicada para o reconhecimento de fonemas. A plataforma escolhida para o desenvolvimento da rede neural artificial CollabNet foi o MATLAB.

4.1.1 Linguagem Shell

A linguagem Shell, também conhecida como linguagem de linha de comando, é uma interface de usuário textual que permite a interação com sistemas operacionais e executar uma variedade de tarefas utilizando comandos e *scripts*. Comumente presente em sistemas Unix-like, como Linux, MacOS e distribuições Unix, a linguagem Shell oferece um meio eficiente de manipular arquivos, gerenciar processos, configurar sistemas e automatizar fluxos de trabalho.

Comandos simples ou sequências complexas podem ser digitados no terminal, permitindo aos usuários controlar o sistema operacional, acessar programas e serviços, além de automatizar tarefas repetitivas através da criação de *scripts*. A versatilidade da linguagem Shell a torna uma ferramenta fundamental para administradores de sistemas, desenvolvedores e qualquer pessoa que busque uma forma poderosa de interagir com um sistema computacional.

4.1.2 Linguagem Python

A linguagem Python é uma linguagem de programação versátil e de alto nível amplamente utilizada para desenvolvimento de software. Reconhecida por sua sintaxe clara e legível, Python é adequada para iniciantes e programadores experientes. Ela oferece uma vasta gama de bibliotecas e *frameworks* que simplificam tarefas comuns, como manipulação de dados, criação de interfaces gráficas e desenvolvimento web.

A abordagem intuitiva e sua natureza interpretada tornam Python uma escolha popular em diversos campos, desde ciência de dados e automação até desenvolvimento de jogos. A versão da linguagem Python versão 3.10.12. Foram empregadas apenas um conjunto fundamental e

limitado de bibliotecas do Python como: Pandas, Matplotlib.pyplot, Os, NumPy, Subprocess, Sklearn, Math, etc. Elas foram essenciais para executar as etapas de pré-processamento.

Com os algoritmos do pré-processamento sendo escrito na linguagem Python, foram utilizadas as bibliotecas Librosa, Soundfile e Scipy para auxiliar no tratamento de dados. A Librosa é um pacote para análise de música e áudio. Ele fornece os blocos de construção necessários para criar sistemas de recuperação de informações musicais. O módulo Soundfile é uma biblioteca de áudio baseada em Libsndfile, CFFI e NumPy. O módulo Soundfile pode ler e gravar arquivos de som. O Scipy é uma biblioteca com funções voltadas para as áreas de matemática, ciência e engenharia.

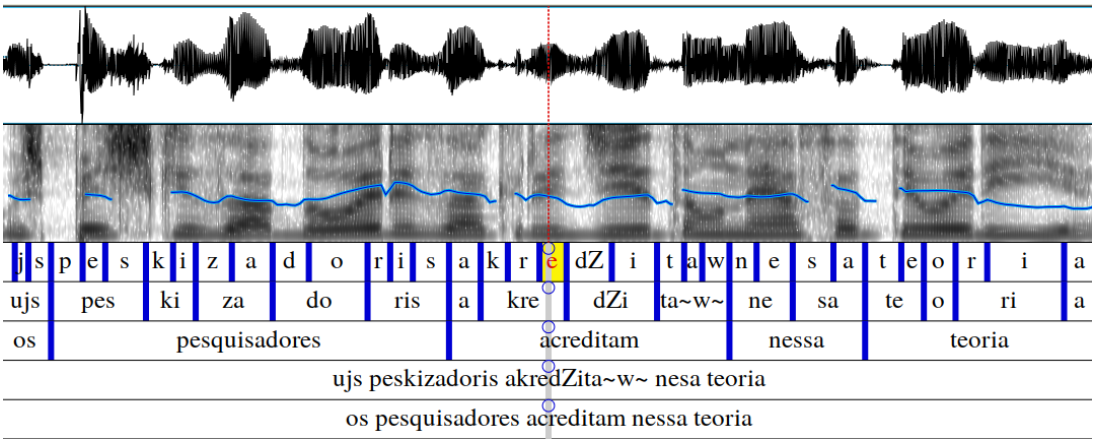
4.1.3 Praat, UFPAlign e Kaldi

O Praat é uma ferramenta de análise de áudios, usado por linguistas, que realiza processos de análise de fala, síntese de fala, classificação, rotulagem e segmentação, experimentos auditivos, manipulação de fala, por meio de métodos estatísticos e de aprendizado de máquina. É um software gratuito e de código aberto. Foi desenvolvido por Paul Boersma e David Weenink na Universidade de Amsterdã. Ele é usado para apresentar os TextGrid do processo de alinhamento fonético. Basicamente serve para ver os resultados do processamento do UFPAlign.

O UFPAlign é uma ferramenta de código aberto para alinhamento fonético automático do Português Brasileiro utilizando o pacote de ferramentas Kaldi. O UFPAlign é disponibilizado como um *plugin* para o Praat, sendo acessível diretamente no menu do Praat e executando o alinhamento a partir de um arquivo de áudio e sua transcrição ortográfica com algumas poucas etapas manuais. O UFPAlign foi desenvolvido pelo grupo FalaBrasil da Universidade Federal do Pará (UFPA).

O UFPAlign fornece um pacote com conversor de grafemas para fonemas (G2P), um sistema de divisão silábica e modelos acústicos. Esses modelos são fundamentados em abordagens estatísticas como GMM (*Gaussian Mixture Models*) para a análise dos elementos linguísticos. Além disso, utiliza HMM-DNN, uma combinação de modelos de redes neurais profundas com modelos markovianos que realizam a identificação de como os componentes da linguagem se relacionam (BATISTA *et al.*, 2022). O resultado é um TextGrid multi-nível contendo fonemas, sílabas, palavras, transcrições fonéticas e ortográficas, conforme a Figura 16.

Figura 16 – Visualização de um arquivo TextGrid através do editor do Praat, incluindo cinco camadas: fonemas, sílabas, palavras, e transcrição fonética e ortográfica, respetivamente.



Fonte: Batista *et al.* (2022)

O Praat é responsável pela representação gráfica da forma de onda do áudio nos domínios do tempo e da frequência, enquanto o Kaldi fornece as marcas temporais para as barras verticais azuis-escuras, apresentadas na Figura 16, que separam os *tokens* uns dos outros.

Adicionalmente, de forma indireta, o Kaldi foi empregado em conjunto com o UFPAlign, integrando-se ao pipeline de processamento. Como uma das principais ferramentas de ASR (Reconhecimento Automático de Fala), reconhecida por atingir o estado da arte para o processamento linguagem natural com os seus modelos, a menção sucinta do Kaldi é relevante devido à sua importância nesse contexto.

O Kaldi é um kit de ferramentas para reconhecimento de fala usado pelo UFPAlign, de código aberto, escrito em C++ e licenciado sob a Apache License v2.0. O Kaldi aspira a ter um código moderno e flexível, de fácil compreensão, modificação e ampliação. Sua estrutura se fundamenta nos transdutores de estados finitos (FST), oferecendo um suporte robusto à álgebra linear, adotando uma abordagem não restritiva (POVEY *et al.*, 2011).

Além disso, disponibiliza de algoritmos genéricos para reutilização e instruções completas para a construção de sistemas de reconhecimento de fala. O principal propósito do Kaldi é facilitar a pesquisa em modelagem acústica (POVEY *et al.*, 2011). O UFPAlign utiliza o Kaldi para realizar a separação em fonemas, sílabas, palavras, transcrições fonéticas e ortográficas (BATISTA *et al.*, 2022).

4.1.4 Google Colaboratory e Jupyter Notebook

As plataformas escolhidas para a etapa do pré-processamento dos dados foi o Google Colaboratory e o Jupyter Notebook. O Google Colaboratory, ou Colab, um serviço em nuvem do Google que permite a execução de código nas linguagens Python e Shell, através do navegador. O Colab é um serviço que não requer configuração para uso e fornece acesso gratuito a recursos de computação, incluindo GPUs (Unidade de Processamento Gráfico), podendo oferecer acesso

a TPUs (*Tensor Processing Unit*) para trabalhos de aprendizado de máquina e aprendizado profundo.

O Jupyter Notebook é uma aplicação popular que permite a criação e compartilhamento de documentos interativos contendo código, visualizações e texto explicativo. Ele oferece um ambiente amigável para programação em várias linguagens, como Python e R, permitindo que os usuários executem trechos de código de maneira incremental e observem os resultados imediatamente. Combinando elementos de código, gráficos e explicações, o Jupyter Notebook é amplamente utilizado em análises de dados, educação e pesquisa, tornando a exploração e comunicação de ideias técnicas mais eficientes e acessíveis.

Vale ressaltar que a execução eficiente do algoritmo está diretamente relacionada à capacidade de processamento, a performance dos equipamentos escolhidos foi de fundamental importância. Devido ao equipamento disponível, um *notebook* com processador Intel Core i5 de décima primeira geração com frequência de 2,4 GHz e uma RAM com oito GB, os algoritmos do pré-processamento foram executados no Colab por disponibilizar 13 GB de RAM, fora a RAM da GPU. O pré-processamento como um todo foi dividido em tarefas e essas tarefas foram divididas em subtarefas.

4.1.5 Base de dados TIMIT

A TIMIT apresenta um conjunto de áudios em língua inglesa, desenvolvida com o propósito de oferecer recursos para pesquisas acústico-fonéticas e para facilitar o avanço e a avaliação de sistemas automatizados de reconhecimento de fala. Essa base de dados inclui gravações de 630 falantes, que abrangem os oito principais dialetos do inglês americano. Cada falante contribuiu com a gravação de dez orações foneticamente ricas, totalizando 6.300 sentenças, equivalentes a cerca de 5,4 horas de áudio. As gravações foram realizadas com resolução de 16 bits e frequências de até 16 kHz (CARVALHO, 2020).

Para cada áudio gravado, foram fornecidas transcrições ortográficas, fonéticas e de palavras, todas elas contendo marcações temporais. Cada um dos falantes recitou dez sentenças, as quais foram agrupadas em três categorias: "sa", "sx"e "si". Dentro desses grupos, duas sentenças pertencem ao grupo "sa", sendo comuns a todos os participantes. Outras cinco sentenças pertencem ao grupo "sx", composto por 450 sentenças foneticamente equilibradas, selecionadas pelo Instituto de Tecnologia de Massachusetts (MIT). As três sentenças restantes se enquadram na categoria "si"e foram escolhidas aleatoriamente de várias fontes, como obras literárias.

Considerando o estado original da base, é possível identificar um total de 64 categorias distintas de fonemas. No entanto, alguns desses fonemas são inaudíveis e podem ser tratados como silêncio, enquanto outros apresentam similaridades consideráveis entre si, permitindo seu agrupamento. Com o intuito de reduzir o número de fonemas, Lee e Hon (1989) propuseram um modelo no qual selecionaram 48 fonemas, excluindo todos os fonemas pertencentes à categoria “q”. Adicionalmente, nesse processo, também foram reconhecidos 15 alofones, que são variações sonoras representando o mesmo fonema.

A TIMIT foi a primeira base usada para visualizar a ideia do centroide como representação do áudio. A Tabela 2 apresenta alguns fonemas, presentes na base TIMIT e exemplos de palavras na língua inglesa em que eles são pronunciados.

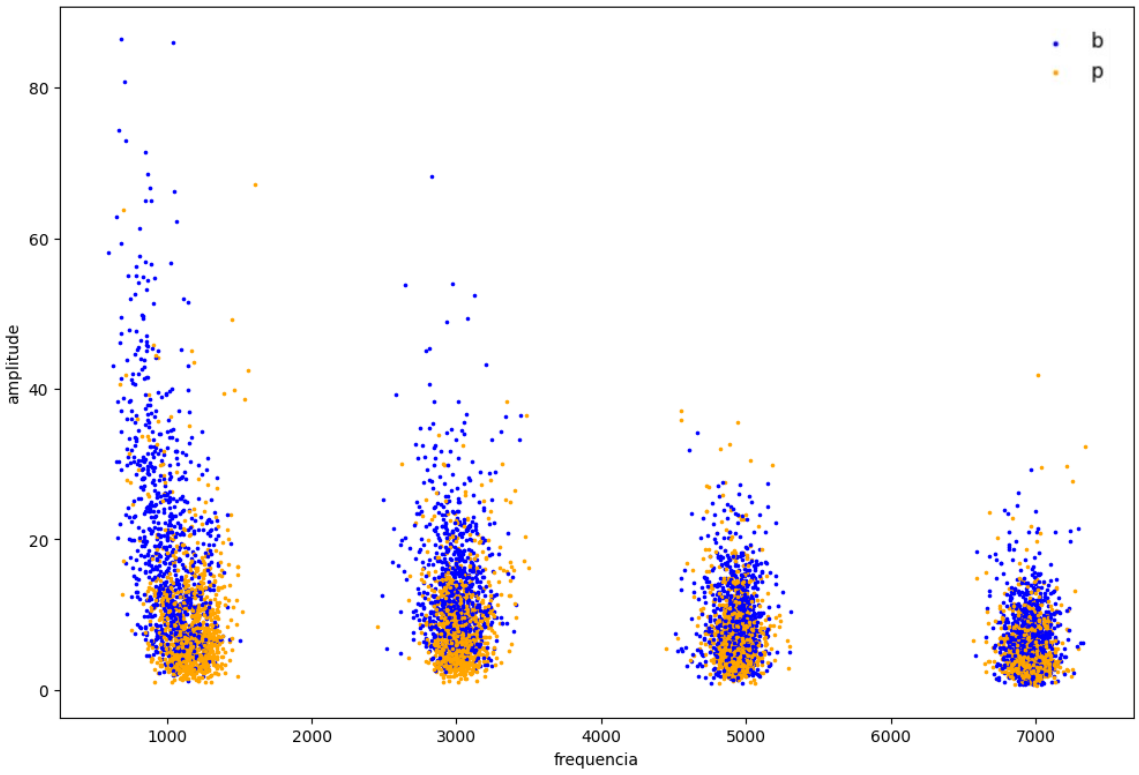
Tabela 2 – Exemplos de fonemas da base *TIMIT* modificada.

Fonema	Palavra
/b/	bob
/p/	pop
/ay/	bite
/ey/	bait
/aw/	bough
/ch/	church

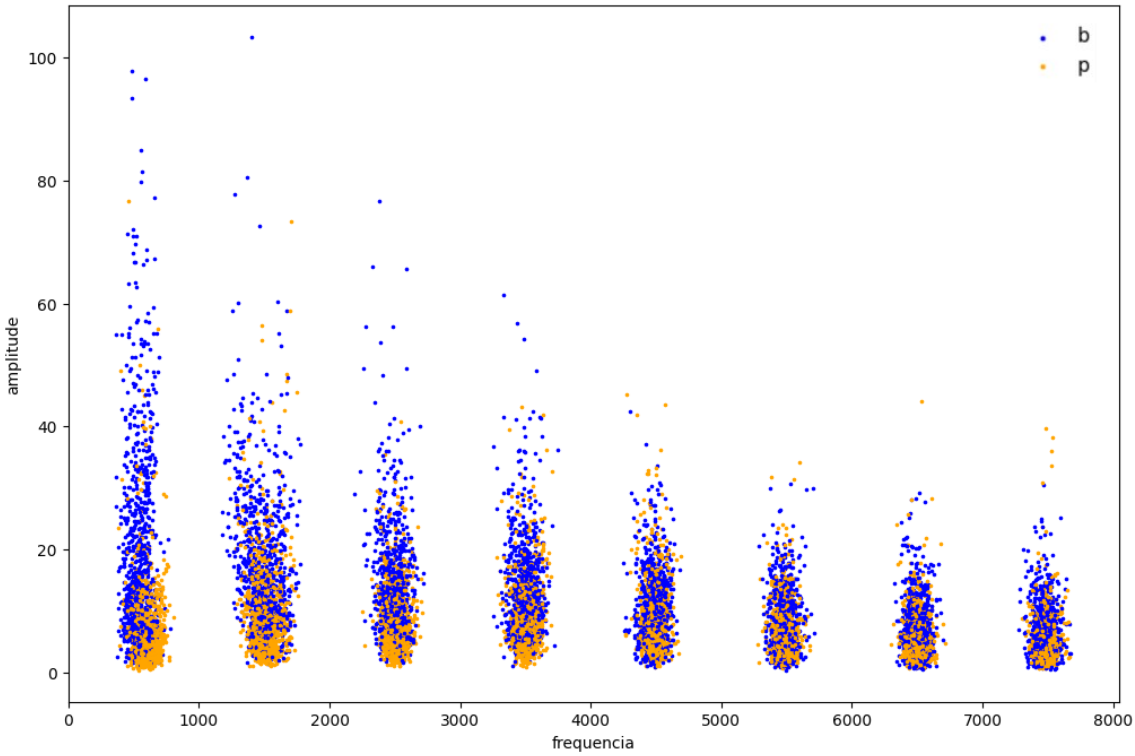
Fonte: Lee e Hon (1989 apud CARVALHO, 2020)

Foram traçados os gráficos dos centroides de fonemas semelhantes, como /b/ e /p/, e diferentes, como /aw/ e /ch/. Na Figura 17, têm-se os gráficos para os fonemas /b/ e /p/. Na Figura 17a, as frequências foram agrupadas em quatro grupos com tamanho de 2000 frequências. Na Figura 17b, as frequências foram agrupadas em oito grupos com tamanho de 1000 frequências. Na Figura 17c, as frequências foram agrupadas em 16 grupos com tamanho de 500 frequências. Pode-se perceber pelos gráficos da Fig. 17, os diferentes agrupamentos de centroides para diferentes quantidades de frequências.

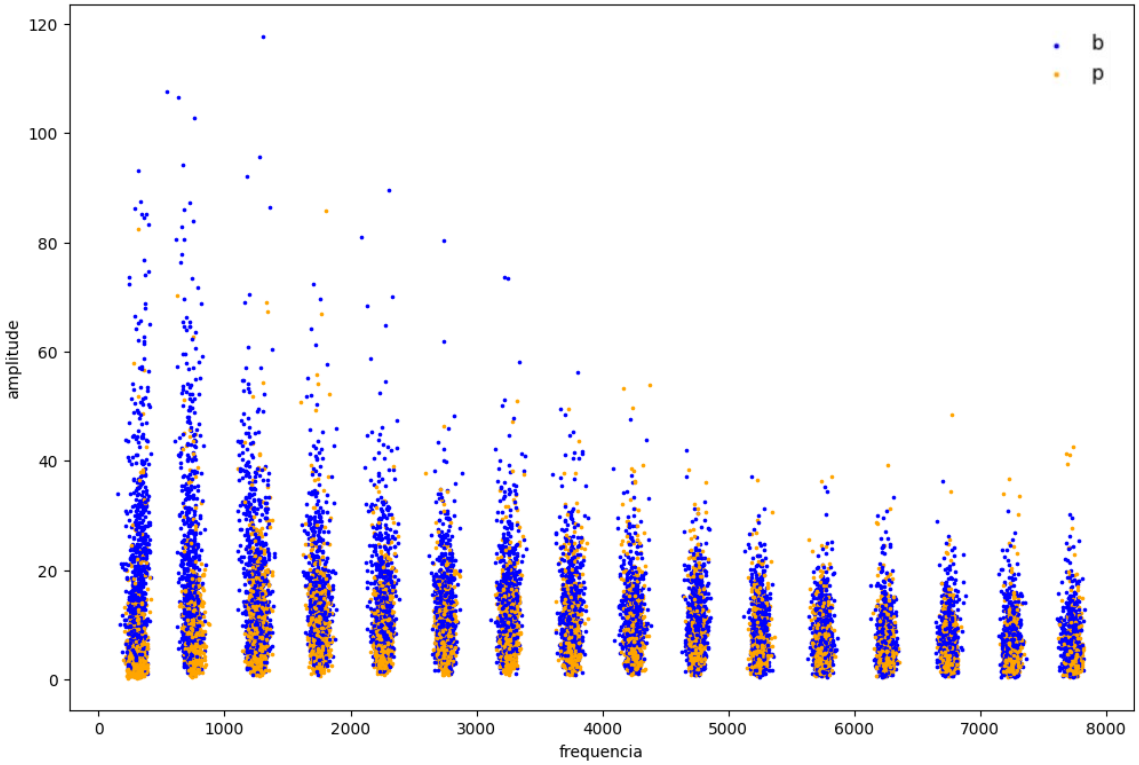
Figura 17 – Fonemas /b/ e /p/ da *TIMIT*.



(a) Amostras de /b/ e /p/ representadas por quatro centroides.



(b) Amostras de /b/ e /p/ representadas por oito centroides.



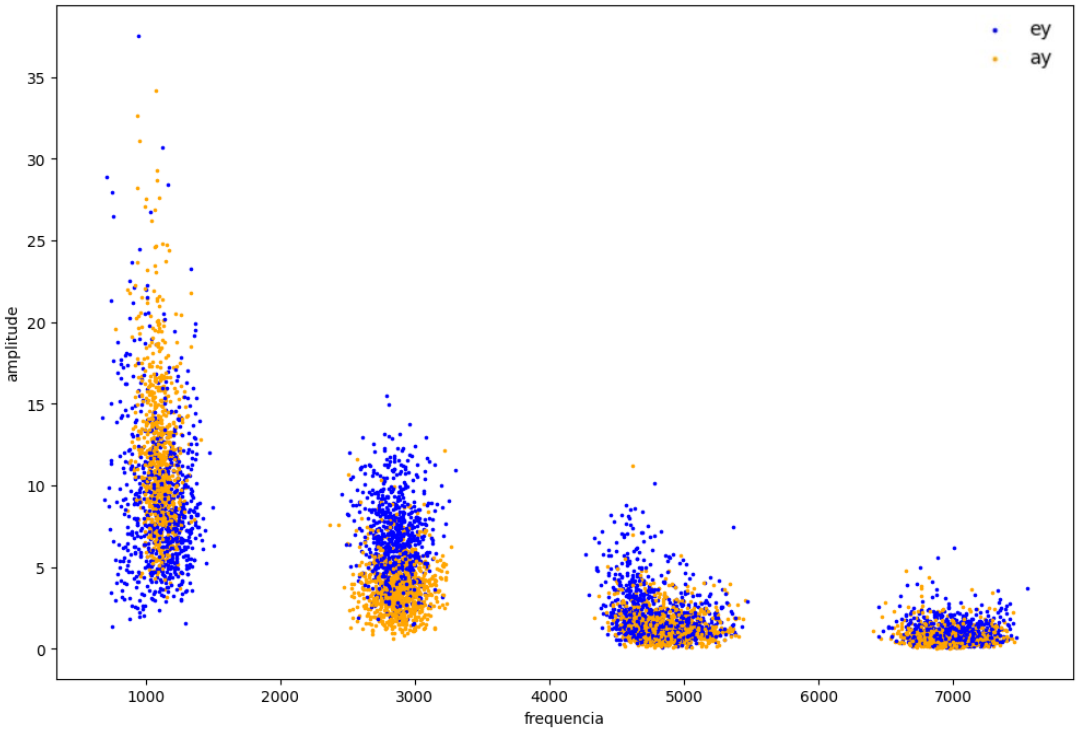
(c) Amostras de /b/ e /p/ representadas por 16 centroides.

Fonte: Elaborado pela autora (2024).

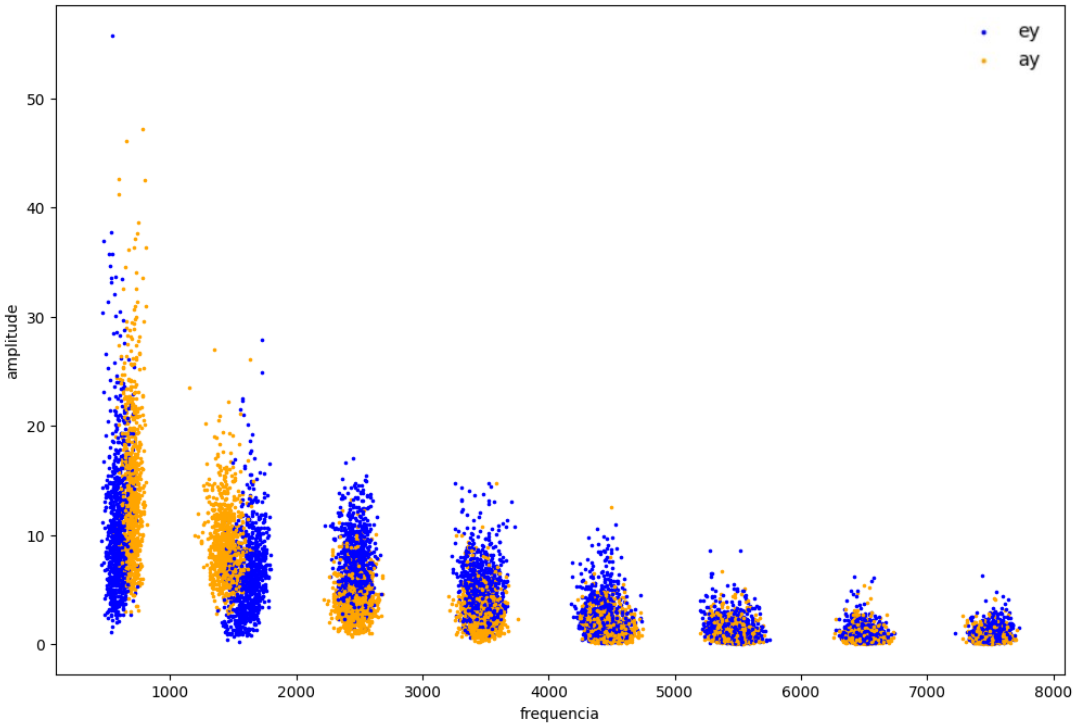
De forma semelhante aos gráficos da Fig. 17, a Figura 18 apresenta os gráficos dos centroides para os fonemas /ay/ e /ey/. Na Figura 18a, as frequências foram agrupadas em quatro

grupos com tamanho de 2000 frequências. Na Figura 18b, as frequências foram agrupadas em oito grupos com tamanho de 1000 frequências. Na Figura 18c, as frequências foram agrupadas em 16 grupos com tamanho de 500 frequências.

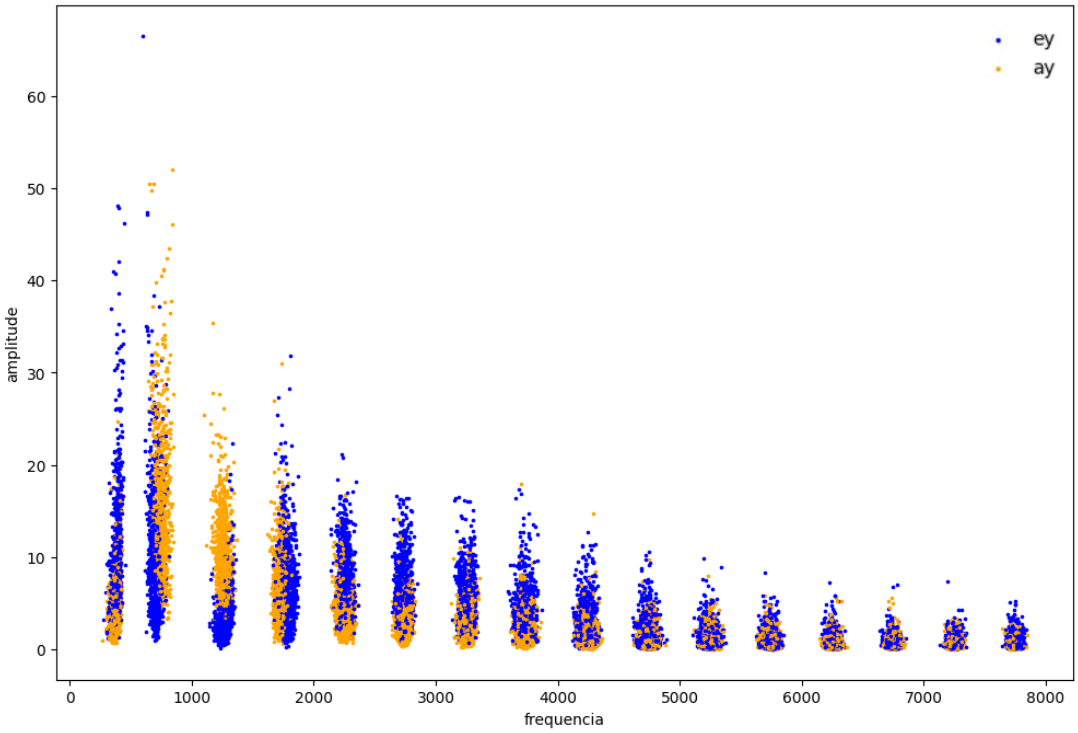
Figura 18 – Fonemas /ay/ e /ey/ da *TIMIT*.



(a) Amostras de /ey/ e /ay/ representadas por quatro centroides.



(b) Amostras de /ey/ e /ay/ representadas por oito centroides.

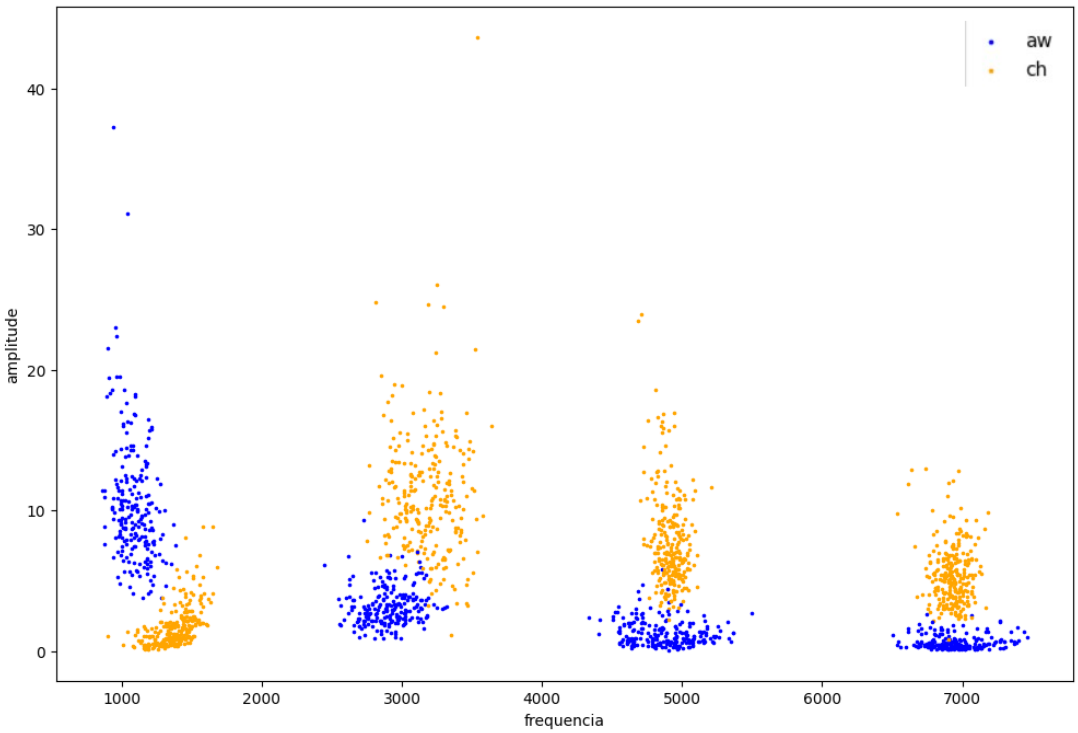


(c) Amostras de /ey/ e /ay/ representadas por 16 centroides.

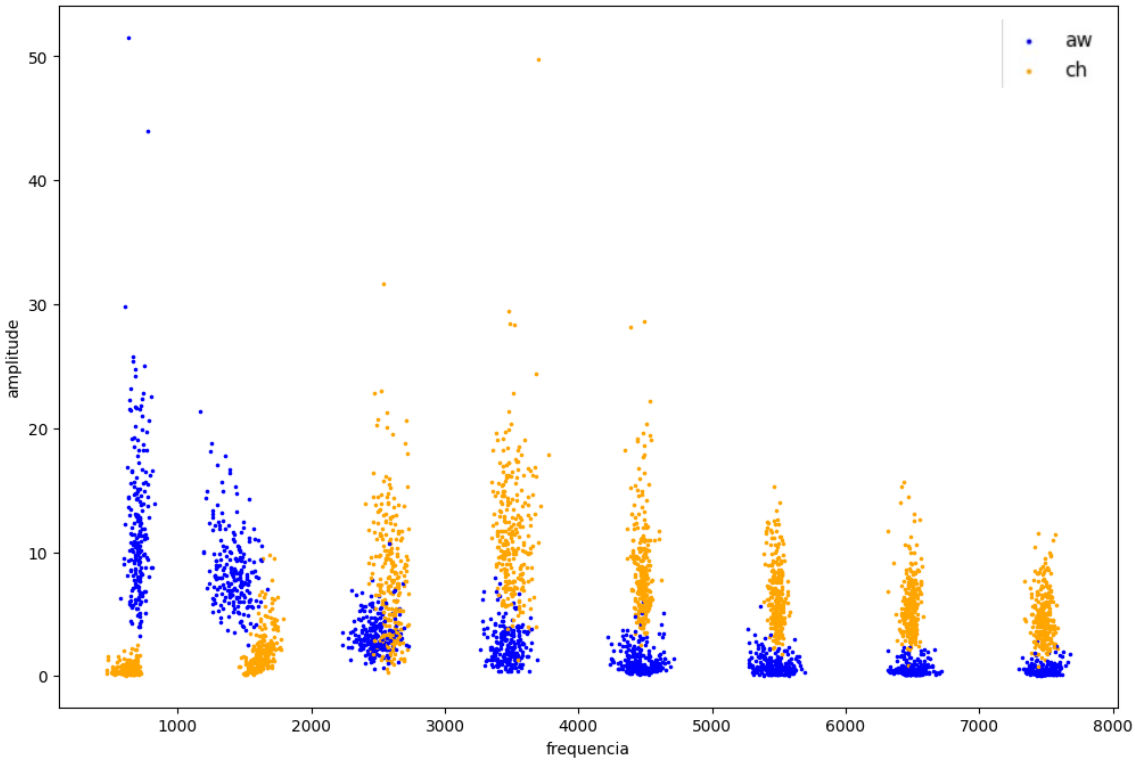
Fonte: Elaborado pela autora (2024).

De forma semelhante aos gráficos das Figuras 17 e 18, a Figura 19 apresenta os gráficos dos centroides para os fonemas /aw/ e /ch/.

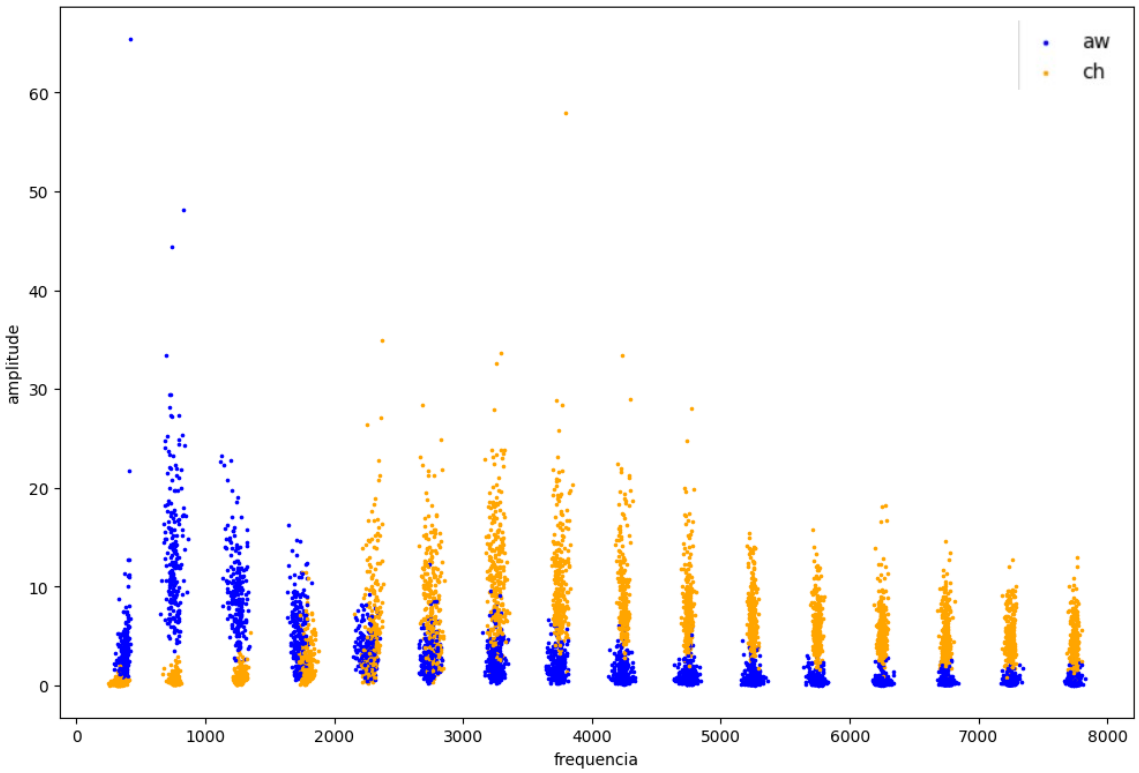
Figura 19 – Fonemas /aw/ e /ch/ da *TIMIT*.



(a) Amostras de /aw/ e /ch/ representadas por quatro centroides.



(b) Amostras de /aw/ e /ch/ representadas por oito centroides.



(c) Amostras de /aw/ e /ch/ representadas por 16 centroides.

Fonte: Elaborado pela autora (2024).

Na Figura 19a, as frequências foram agrupadas em quatro grupos com tamanho de 2000

frequências. Na Figura 19b, as frequências foram agrupadas em oito grupos com tamanho de 1000 frequências. Na Figura 19c, as frequências foram agrupadas em 16 grupos com tamanho de 500 frequências. Com isso, foi observado que fonemas parecidos, como /b/ e /p/, apesar de suas semelhanças, ainda conseguem ser agrupados separadamente. E fonemas diferentes, como /ey/ e /ay/ ou /aw/ e /ch/, conseguem ser agrupados sem tanta dificuldade.

4.1.6 Base de dados CSLU: Spoltech Brazilian Portuguese Version 1.0

A CSLU: Spoltech Brazilian Portuguese Version 1.0 contém 480 falantes de diversas regiões do Brasil e 8.207 enunciados distintos. Um total de 2.540 enunciados foram transcritos no nível de palavra (sem alinhamento temporal) e 5.479 enunciados foram transcritos no nível de fonema (com alinhamento temporal) (SAMPAIO NETO *et al.*, 2008). A elaboração do protocolo, a gravação e a transcrição foram realizadas pela Universidade Federal do Rio Grande do Sul (UFRGS) e pela Universidade de Caxias do Sul .

Para cada amostra de áudio na Spoltech, há um documento em texto o qual contém anotações relativas à posição temporal de cada fonema presente no enunciado. Dessa maneira, é possível construir um algoritmo que recebe as informações temporais de início e fim de cada fonema e posteriormente separar os áudios em fonemas, sílabas fonéticas, sílabas ou palavras e agrupá-los de acordo com a sua classificação fonética (SAMPAIO NETO *et al.*, 2008).

Inicialmente a ideia era trabalhar com a base Spoltech. Acreditava-se que seria uma base interessante devido a variabilidade dos locutores (falantes) e das orações (enunciados) ditas. Todavia, houve demora no acesso a Spoltech por ser uma base de dados privada. Somente após os proprietários disponibilizarem essa base gratuitamente, foi realizado o acesso e uma análise aos dados da Spoltech.

Embora útil, o corpus Spoltech possui vários problemas. Percebeu-se tratar-se de uma base não saneada. O ambiente de gravação não foi controlado. Algumas gravações foram feitas em estúdio e outras em ambientes ruidosos. A base é incompleta e irregular: apesar de dividida em transcrições, fonemas e áudios, possui áudios ausentes, transcrições ausentes, até mesmo erros gráficos nas transcrições. Outro aspecto problemático é que suas transcrições possuem muitos erros. Isso dificultou o alinhamento e corte dos fragmentos. O trabalho de Sampaio Neto (2011) confirma que uma trabalhosa e manual correção dos arquivos de áudio e texto precisa ser realizada. Embora feito o levantamento da base, chegou-se a conclusão que, por possuir incorreções, ela não estava adequada para uso.

4.1.7 Base de dados LaPS Benchmark 16k

A LaPS Benchmark 16k é um corpus composto por 700 orações pronunciadas por 25 homens e 10 mulheres. Esse conjunto totaliza aproximadamente 54 minutos de gravações de áudio. Todas as gravações foram realizadas em computadores utilizando microfones comuns. A taxa de amostragem utilizada foi de 16.000 Hz, com cada amostra representada por 16

bits de resolução. É importante notar que o ambiente de gravação não foi controlado e ruídos estavam presentes nas gravações, com o objetivo de caracterizar ambientes reais onde sistemas de reconhecimento de voz são utilizados (SAMPAIO NETO, 2011).

4.1.8 Base de dados Male/Female for Forced Phonetic Alignment

A Male/Female for Forced Phonetic Alignment é um conjunto de dados de dois falantes alinhados manualmente no nível do fonema (arquivos Textgrid de Praat fornecidos). Cada porção contém 200 ocorrências de um locutor masculino e feminino. Ela consiste em uma base de dados pequena, construída em estúdio, com áudios limpos e de boa qualidade e que já foi saneada, o que a torna ideal para o processo de treinamento.

4.1.9 MATLAB

O MATLAB é um ambiente de computação numérica desenvolvida pela MathWorks e possui uma linguagem de programação multiparadigma própria. O MATLAB permite manipulações matriciais, traçado de funções e dados, implementação de algoritmos, criação de interfaces de usuário e interface com programas escritos em outras linguagens.

4.2 MÉTODOS

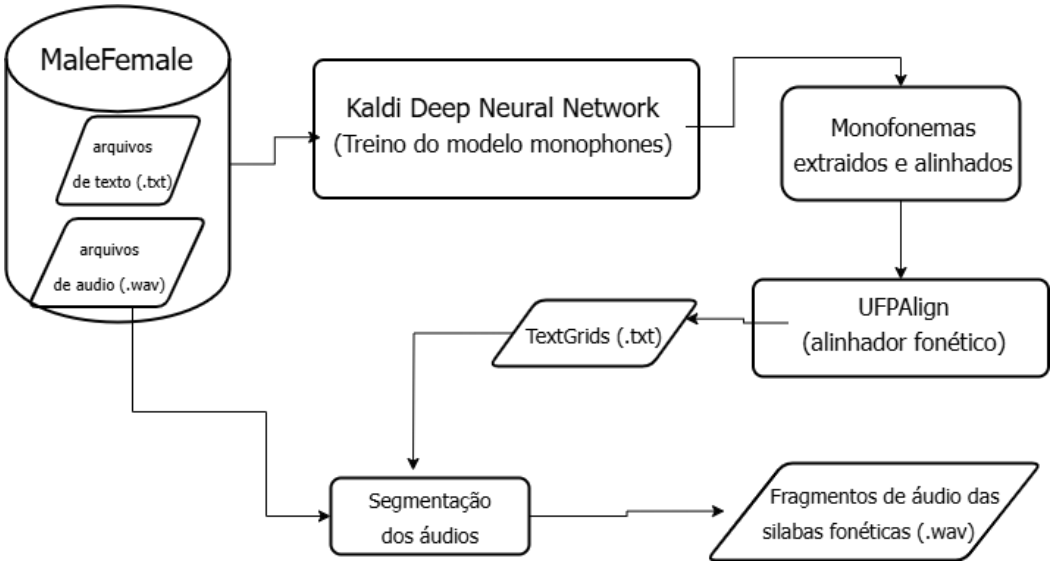
4.2.1 Pré-processamento dos dados

Para a preparação da base foi utilizado o UFPAlign, que é uma uma ferramenta para alinhamento automático de áudio e texto em português do Brasil. Esse alinhador foi desenvolvido e disponibilizado como um *plugin* para o Praat. O UFPAlign funciona em sistemas operacionais Linux.

Quando o alinhamento é concluído com sucesso, o alinhador oferece a opção de exibir imediatamente o TextGrid resultante na interface do Praat ou prosseguir para alinhar um novo arquivo de áudio. O TextGrid multicamadas resultante de alinhamento contém cinco camadas: fonemas, sílabas fonéticas, palavras, transcrição fonética e transcrição ortográfica, respectivamente.

Após o entendimento obtido sobre isso, com o objetivo de automatizar o processo de uso do UFPAlign, apresentado na Figura 20, foi criado um *script* utilizando as linguagens Python e Shell, a biblioteca Os, os arquivos de aúdio e os arquivos de texto da base de dados para gerar esses TextGrids.

Figura 20 – Uso do UFPAlign para realizar o alinhamento forçado das sílabas fonéticas.

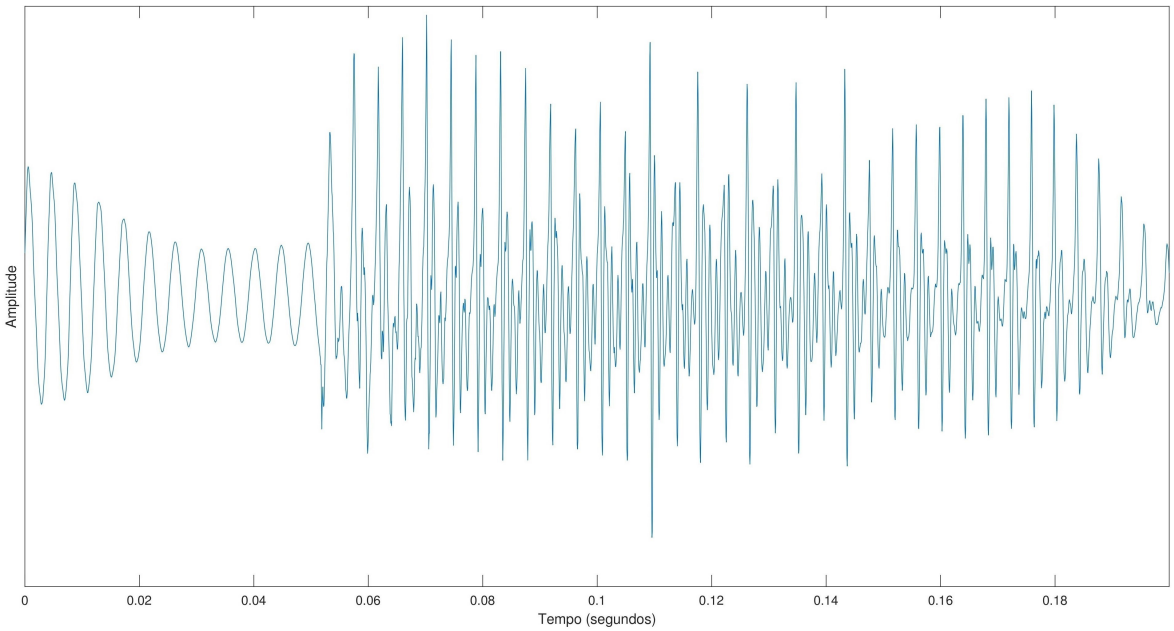


Fonte: Elaborado pela autora (2024).

Em seguida, após finalizado o alinhamento, um outro *script* foi criado utilizando as linguagens Python e Shell, as bibliotecas Librosa, Soundfile, Scipy e Os, os arquivos de áudio da base de dados e os seus respectivos TextGrids com o objetivo de realizar os cortes dos áudios a nível de sílaba fonética de cada palavra, tendo assim os fragmentos de áudio.

Posteriormente, um outro *script* foi criado utilizando as linguagens Python e Shell, as bibliotecas Librosa, Soundfile e Scipy e os fragmentos de áudio. Os fragmentos foram lidos em formato .wav e apresentados no domínio do tempo, conforme a Figura 21.

Figura 21 – Gráfico da sílaba fonética “da” no domínio do tempo.

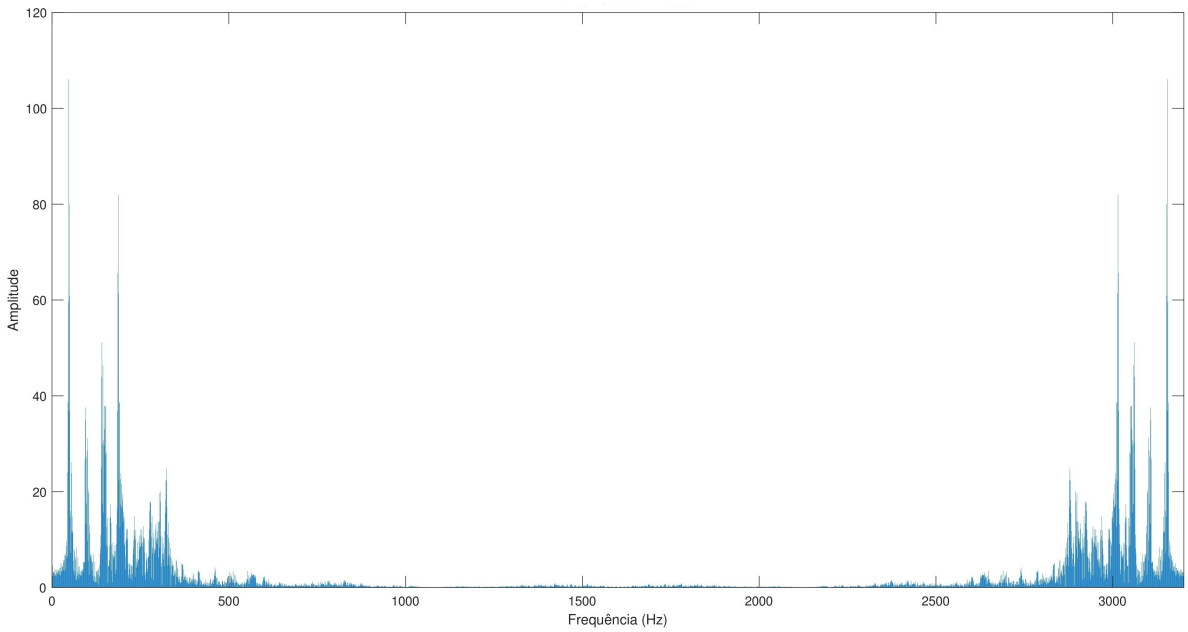


Fonte: Elaborado pela autora (2024).

Após todos os arquivos lidos, foi calculada a *Fast Fourier Transform* (FFT), conforme a Equação 4.1, de cada arquivo para se obter o áudio no domínio da frequência, conforme a Figura 22.

$$\mathscr{F}[x(t)] = X(\omega) = \int_{-\infty}^{\infty} x(t) \times e^{-j\omega t} dt \tag{4.1}$$

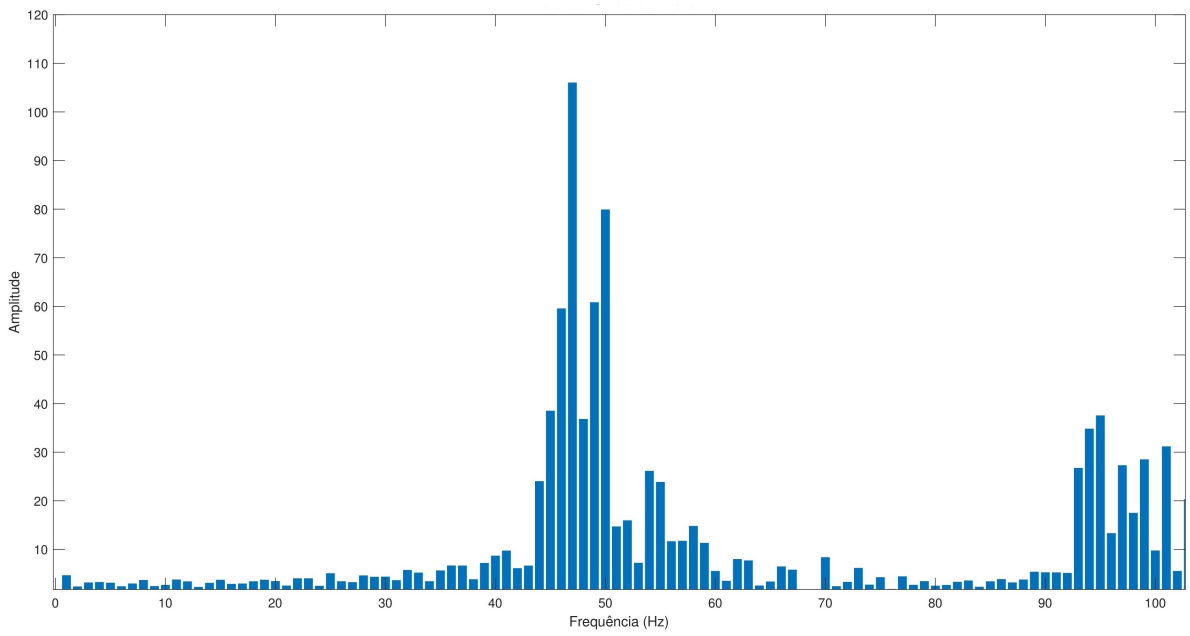
Figura 22 – FFT do sinal sonoro da sílaba fonética “da”.



Fonte: Elaborado pela autora (2024).

Para melhor visualização, a Fig. 23 apresenta uma ampliação das frequências iniciais do gráfico da Fig. 22.

Figura 23 – Aproximação do gráfico até 100 dados amostrados.

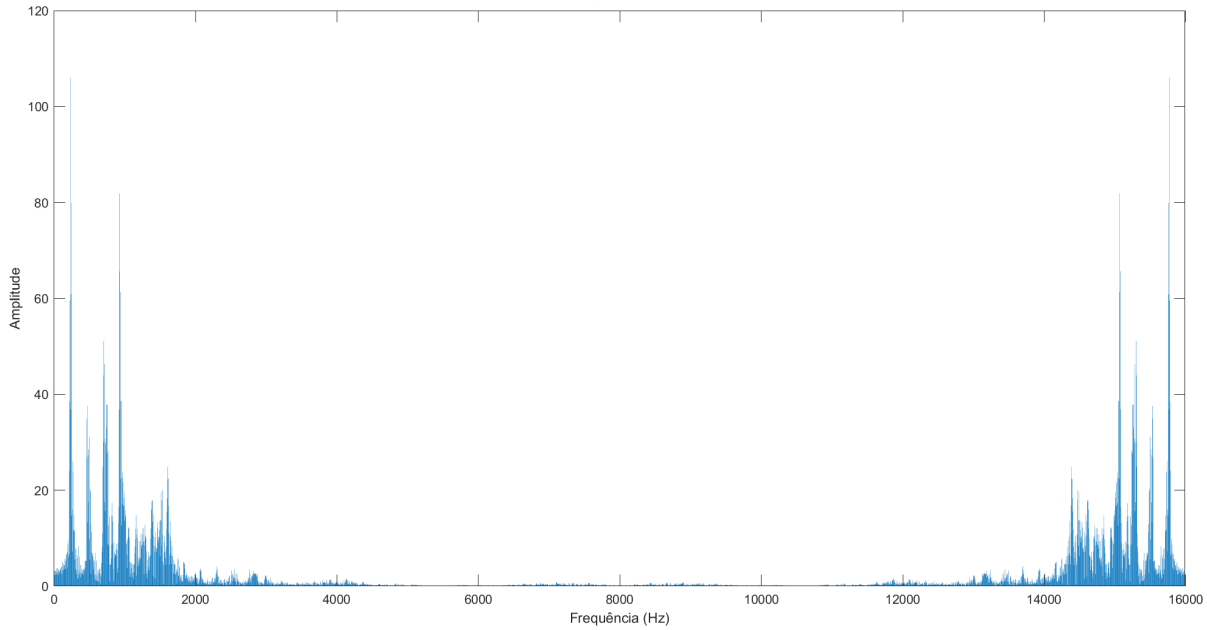


Fonte: Elaborado pela autora (2024).

Foi observado que alguns sinais sonoros alcançam diferentes frequências. Buscando solucionar este problema, foi aplicado o teorema de Nyquist, o qual afirma que para converter um sinal analógico em sinal digital semelhante e viável em relação o sinal original, a frequência da taxa de amostragem deve ser no mínimo duas vezes maior que a frequência do sinal de interesse.

Outra informação importante é que o ouvido humano é capaz de captar sons entre 20 Hz e 20 kHz. Também se observou no estado da arte que os sinais sonoros eram amostrados por meio de uma frequência de amostragem de 32 kHz. Logo foi aplicado uma frequência de amostragem de 32 kHz para que a frequência do sinal seja igual a 16 kHz. Com isso, todos os sinais, independente das classes, estão amostrados igualmente, conforme a Figura 24.

Figura 24 – FFT do sinal sonoro amostrado da sílaba fonética “da”.



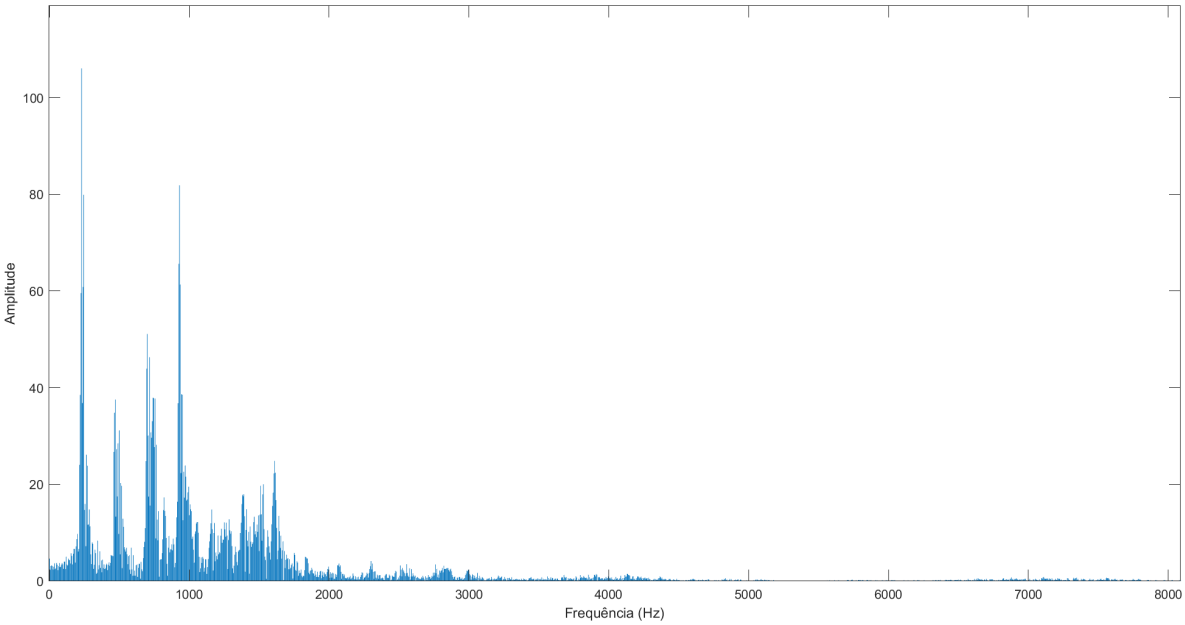
Fonte: Elaborado pela autora (2024).

As informações sobre esses arquivos foram salvas em um arquivo CSV com objetivo de otimizar a leitura para geração de centroides. No CSV, foram salvos a sílaba, o tempo de duração, os dados no domínio da frequência, informações sobre o locutor, entre outros.

É sabido que representar sinais no domínio da frequência simplifica a análise matemática. Analisar a frequência de um sinal ou sistema possibilita obter informações preciosas sobre suas características, reconhecer padrões, detetar anomalias e, ainda, criar estratégias mais eficazes para explorar os dados.

A FFT proporciona um sinal refletido no seu espectro de frequência. Conforme o estado da arte e com o objetivo de reduzir custo computacional, foi considerado os dados da metade do espectro de frequência, ou seja, até 8 kHz conforme a Figura 25.

Figura 25 – Metade da FFT do sinal sonoro amostrado do fonema “da”.



Fonte: Elaborado pela autora (2024).

Estas barras foram consideradas como objetos geométricos conforme as Figuras 25 e 26. Esses objetos foram agrupados e para representar as frequências desses gráficos, foi utilizado um suposto centroide que corresponde ao centro geométrico de uma figura plana.

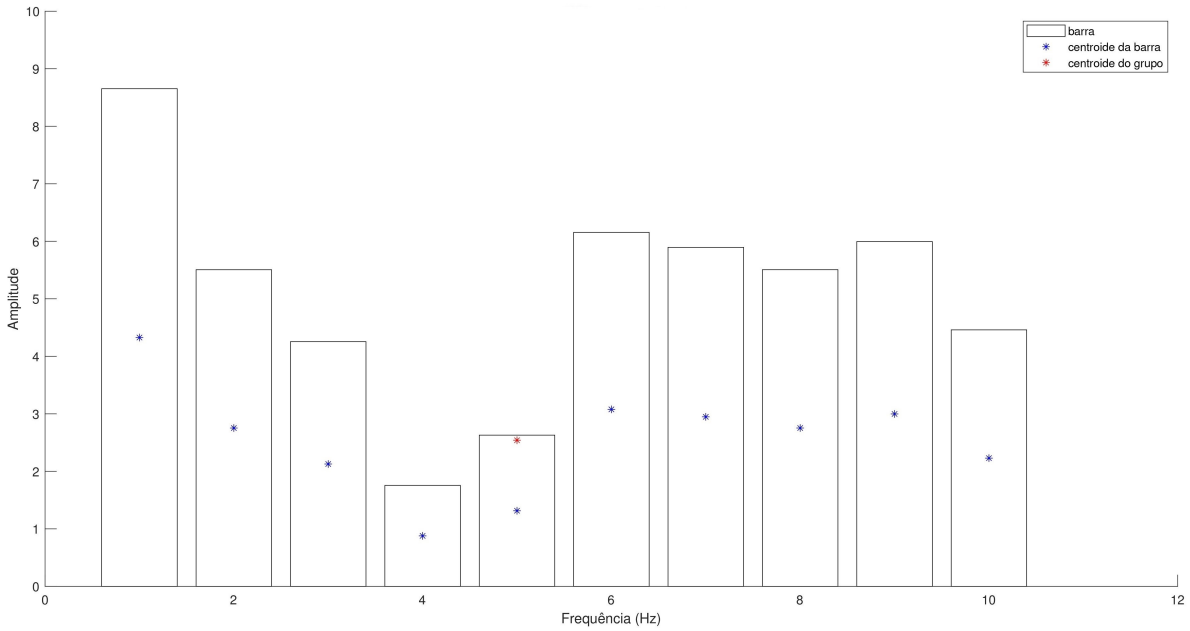
Com os dados de áudio no domínio da frequência, foram realizados agrupamentos, onde cada grupo foi representado por um centroide, conforme as Equações 4.2, 4.3 e 4.4, para cada faixa de frequência pré-determinada de maneira que fique espaçado igualmente. Por exemplo, em um sinal com uma faixa de frequência de 8 kHz, sendo feito um centroide a cada 10 Hz, obtêm-se um total de 800 centroides (800 grupos com 10 barras em cada um). Na Figura 26, tem-se um grupo do exemplo citado acima. Esta operação visa a diminuir o custo computacional sem comprometer muito o aprendizado da rede.

$$C_M = (X_M, Y_M) \tag{4.2}$$

$$X_M = \frac{\sum_{f=1}^{n} x_f \times a_f}{\sum a_f} \tag{4.3}$$

$$Y_M = \frac{\sum_{f=1}^{n} \frac{y_f}{2} \times a_f}{\sum a_f} \tag{4.4}$$

Figura 26 – Grupo com 10 Hz, no qual foi determinado um centroide para representá-lo.



Fonte: Elaborado pela autora (2024).

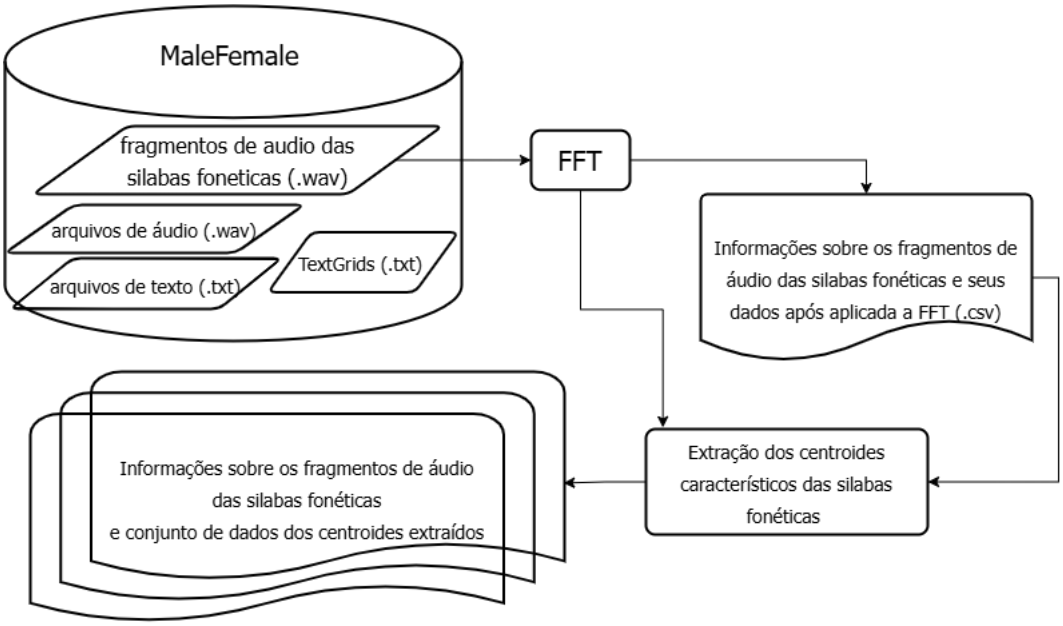
Nas Equações 4.2, 4.3 e 4.4, x_f é a posição da frequência após a FFT, y_f sua magnitude após a FFT, X_m é a posição do centroide na faixa de frequência e Y_m a posição do centroide na faixa da magnitude. O centroide precisa que o termo a_f corresponda a área do objeto, no entanto devido à transformada do objeto as barras são verticais com mesma dimensão, mudando apenas as coordenadas da frequência e amplitude, dessa forma não seria possível calcular a área da barra vertical, pois a transformada não gera um retângulo e sim valores reais, então consideramos o termo a_f igual à y_f. Dessa forma as coordenadas do centroide do agrupamento ficam:

$$X_M = \frac{\sum_{f=1}^{n} x_f \times y_f}{\sum y_f} \tag{4.5}$$

$$Y_M = \frac{\sum_{f=1}^{n} \frac{y_f^2}{2}}{\sum y_f} \tag{4.6}$$

Foram calculados dois, quatro, oito, dez, 16, 80, 160, 400 e 800 centroides para representar as amostras de áudio e para otimizar o algoritmo, foi gerado um arquivo CSV para cada conjunto de dados de centroides, que guarda as informações dos fragmentos de áudio (sílaba, nome do arquivo, locução, instante inicial, instante final e duração) e as coordenadas dos centroides calculados. A Figura 27 apresenta todo o processo descrito após o alinhamento forçado das sílabas fonéticas até a geração desses arquivos CSVs.

Figura 27 – Fase de extração de características.



Fonte: Elaborado pela autora (2024).

Após o pré-processamento, foi utilizada uma rede neural *deep stacked autoencoder*. A rede utilizada é denominada CollabNet, apresentada no Capítulo 5. Como dados de entrada para a rede, foram utilizadas as coordenadas do centroide. No entanto, antes de serem inseridos na rede, os dados foram normalizados utilizando o Min-Max, a fim de diminuir a distância entre os valores das coordenadas muito espaçadas, reduzindo, consequentemente, a influência causada por valores muito altos em relação aos demais. Para que isso ocorra, foi utilizada a Equação 4.7:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{4.7}$$

5 COLLABNET - *DEEP FEEDFORWARD* COLABORATIVA

Este capítulo define a rede neural que foi utilizada neste trabalho. A Seção 5.1 mostra o contexto e o princípio aos quais a CollabNet foi criada. Na Seção 5.2, apresenta-se a estrutura da rede. Na Seção 5.3, são expostos detalhes relevantes de como promove-se o treinamento com a CollabNet, explanando mais sobre a máscara D e a variável c.

5.1 CONTEXTUALIZAÇÃO

Conforme observado, redes neurais profundas são principalmente identificadas pela presença de múltiplas camadas ocultas. É conhecido também que a topologia da rede sempre foi um dos principais fatores determinantes para o sucesso desses algoritmos (GLOROT; BENGIO, 2010). No entanto, durante o processo de inserção de novas camadas ocultas, muitas arquiteturas recorrem a métodos estocásticos para inicializar os pesos dos neurônios nessa camada, o que pode resultar em perturbações no erro de treinamento.

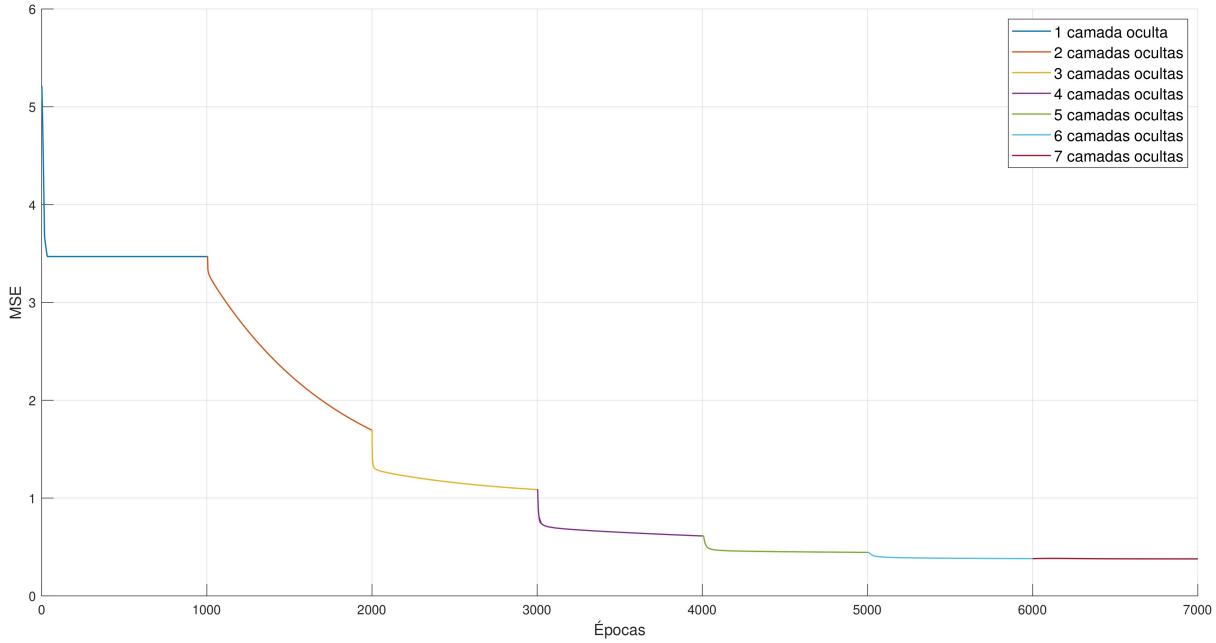
Assim, com o objetivo de aprimorar o processo de inserção de novas camadas ocultas em redes neurais, com base nos conceitos de *autoencoders* e redes *deep stacked autoencoders*, Lima Junior (2018) apresentou uma abordagem denominada de CollabNet.

Na obra de Lima Junior (2018), foi implementado um esquema para adicionar camadas ocultas em uma rede neural do tipo *feedforward*, a fim de evitar as inconsistências no erro de saída no decorrer do treinamento, que podem surgir ao inserir uma nova camada usando os métodos convencionais. Era crucial garantir uma integração adequada entre as novas camadas, com o propósito do erro de saída da rede sempre convergir. Essa convergência constante precisa ser de tal forma, que cada nova camada melhore o desempenho das camadas anteriores, permitindo uma colaboração efetiva entre elas.

Assim sendo, Lima Junior (2018) procurou uma abordagem eficiente para aumentar a profundidade da rede, permitindo a adição de camadas escondidas de forma colaborativa durante a execução do software.

A abordagem adotada neste método se baseia no princípio de que cada nova camada deve iniciar, imediatamente, o treinamento exatamente do ponto em que a camada anterior parou. Ou seja, o objetivo é garantir uma redução gradual do erro, mesmo quando uma nova camada é adicionada à rede neural, conforme ilustrado na Figura 28. Essa figura, apresenta o erro quadrático médio (MSE, em inglês) de saída de uma rede neural. O gráfico da Figura 28 foi obtido com um exemplo de sucesso. Essa estratégia tem a intenção de promover um aprendizado progressivo, evitando quedas em mínimos locais ou platôs que possam prejudicar o desempenho da rede.

Figura 28 – Comportamento do erro esperado.



Fonte: Elaborado pela autora (2024).

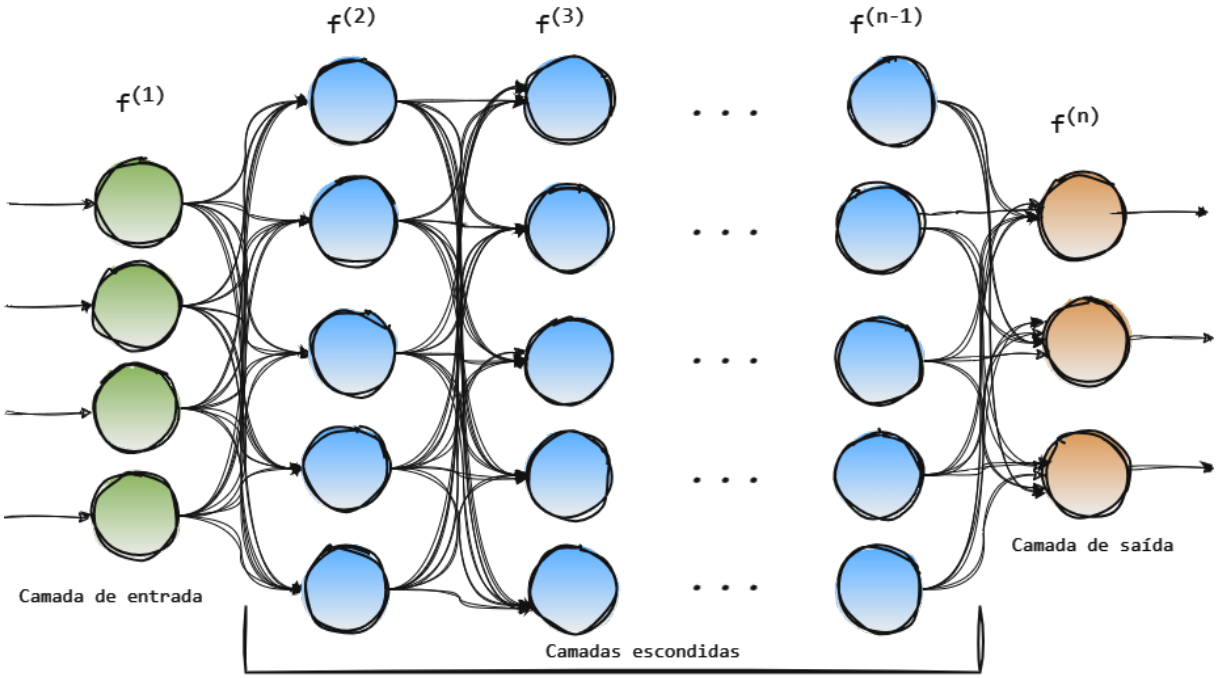
No entanto, para garantir o sucesso dessa abordagem, é crucial realizar a integração adequada da nova camada. Caso contrário, a adição da nova camada pode ter um impacto negativo na aprendizagem já alcançada, resultando em uma deterioração no desempenho geral.

Portanto, a principal ideia dessa proposta foi desenvolver uma técnica que permita incorporar o conhecimento adquirido pelas camadas anteriores no treinamento da nova camada. Vale ressaltar que essa nova camada oculta inserida deve possuir o mesmo número de neurônios da camada anterior, bem como preservar o aprendizado obtido por ela. Dessa forma, o objetivo é fornecer à nova camada informações sobre o aprendizado já realizado pela rede, a fim de aproveitar esse conhecimento prévio durante o treinamento.

5.2 ESTRUTURA

Essa proposta pretende aprimorar a estrutura inicial da CollabNet por meio da adição gradual de camadas escondidas. Cada nova camada é inserida de forma sequencial, permitindo que o aprendizado já adquirido seja preservado. Para garantir isso, um pré-processamento é aplicado à camada recém inserida, com o objetivo de incorporar progressivamente a influência dessa camada no treinamento global da rede. Essa etapa de pré-processamento é essencial para evitar que as novas camadas tenham um impacto negativo no aprendizado geral, evitando o aumento do erro. Ao final das inserções, a estrutura da CollabNet é similar à apresentada na Figura 29.

Figura 29 – Estrutura básica da CollabNet.



Fonte: Elaborado pela autora (2024).

Percebe-se que na estrutura, conforme a Figura 29, as camadas devem seguir o número de neurônios estabelecidos, com exceção da camada de entrada e da camada de saída. Esse princípio é fundamental para o funcionamento da técnica, sobretudo durante o treinamento.

Na Equação 5.1, pode-se visualizar a representação da estrutura da CollabNet, onde w representa os pesos de treinamento das camadas. As funções de ativação dos neurônios são denotadas por $f^{(1)}$ para a primeira camada, $f^{(2)}$ para a segunda camada e assim por diante, com a notação $f^{(n-1)}$ sendo utilizada para as demais camadas. A última camada é chamada como a camada de saída da rede e tem por função de ativação $f^{(n)}$. Segundo Goodfellow *et al.* (2016) a profundidade da rede é determinada pelo comprimento total da cadeia de camadas.

$$f(x) = w \cdot f^{(3)}(w \cdot f^{(2)}(w \cdot f^{1}(x))) \tag{5.1}$$

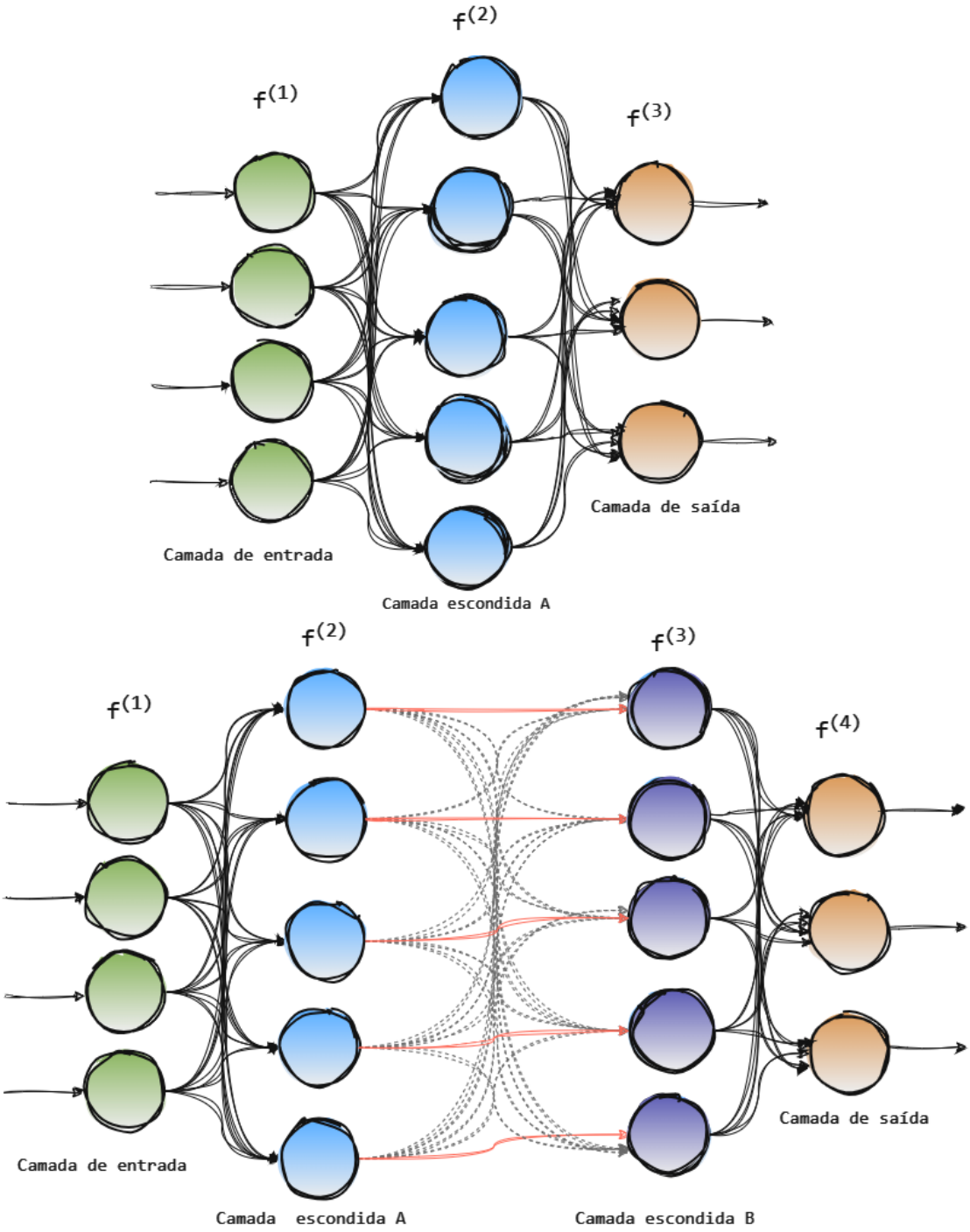
5.3 TREINAMENTO

O modo de treinamento utilizado neste trabalho começa de forma semelhante ao método convencional de uma rede MLP, com apenas uma camada escondida e usando o algoritmo *backpropagation*, conforme proposto por Glorot e Bengio (2010). Após essa fase inicial de treinamento, quando o erro de saída da rede não diminui mais, é possível adicionar uma nova camada à rede em treinamento. No entanto, essa inserção de uma nova camada requer vários procedimentos para garantir uma integração harmoniosa e bem-sucedida das novas camadas escondidas.

O método proposto tem como principal objetivo assegurar que a adição de novas camadas altere o erro de saída da rede. Para alcançar esse objetivo, é crucial que a saída do neurônio na

camada recém inserida seja idêntica à saída do neurônio correspondente na camada anterior, levando em conta todos os dados de entrada. Essa condição de igualdade é ilustrada na Figura 30, que representa visualmente a estrutura da rede com a nova camada e suas conexões.

Figura 30 – Inserção de uma nova camada.



Fonte: Elaborado pela autora (2024).

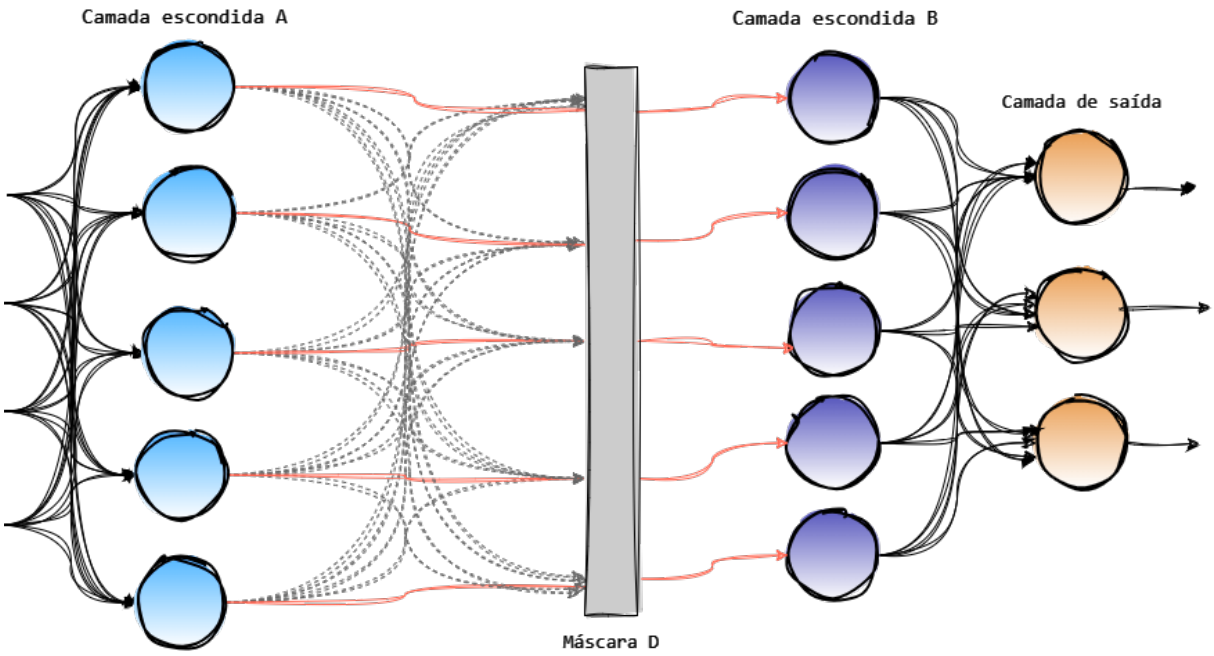
5.3.1 Máscara D

Neste método, a inicialização dos pesos entre as camadas A e B ocorre de modo aleatório. Contudo, para assegurar que a saída da camada B seja idêntica à saída da camada A, foi preciso aplicar um tratamento na saída da camada A. Caso contrário, não há garantia de que a saída da camada B é igual à saída da camada A, já que os pesos recém-inicializados e a função de ativação dos neurônios na camada B podem alterar essa saída.

Para mitigar o impacto causado pelos pesos recém criados, este trabalho faz um tratamento dos dados entre as camadas A e B. Esse tratamento é realizado por meio de uma máscara, designada como *D*, como demonstrado na Figura 31.

Essa máscara *D* é fundamental para estabelecer a equivalência entre as saídas das duas camadas e preservar o aprendizado prévio da rede. Ao modificar os dados que chegam à camada B, a máscara *D* permite que a nova camada se ajuste gradativamente, evitando perturbações drásticas no processo de treinamento e garantindo uma integração harmoniosa das novas camadas. Com essa abordagem, é possível garantir que a adição de camadas não corrompa o aprendizado anterior, resultando em uma estrutura de rede mais robusta e capaz de lidar com tarefas complexas.

Figura 31 – Máscara *D* entre uma camada antiga e a camada recém inserida.



Fonte: Elaborado pela autora (2024).

A máscara *D* desempenha um papel fundamental ao garantir que cada neurônio da nova camada receba exclusivamente a influência do seu neurônio correspondente na camada anterior. Na Figura 31, é possível observar seu funcionamento. As conexões representadas por linhas vermelhas indicam que o valor não sofre alteração pela máscara. Em outras palavras, os

neurônios da nova camada inserida (B) recebem exatamente o mesmo valor de saída do neurônio correspondente na camada anterior (A).

Por outro lado, as conexões representadas por linhas cinzas têm seus valores anulados pela máscara *D*, fazendo com que o valor recebido pelos neurônios da nova camada seja nulo em relação aos neurônios adjacentes. Essa configuração permite que o valor de entrada de cada neurônio na nova camada seja exatamente igual ao valor que sai do neurônio correspondente na camada A.

A máscara *D* é fundamental para a abordagem proposta, pois realiza um filtro através da multiplicação pelo inverso do valor do peso correspondente. Isso garante que o valor de entrada dos neurônios na nova camada (B) seja exatamente igual ao valor de saída do neurônio correspondente na camada anterior (A). Com esse processo, o treinamento da camada B inicia-se a partir do ponto onde a camada A parou, preservando o aprendizado anterior.

Uma vez inserida a camada B, realiza-se o cálculo da entrada desta camada, como mostra a Equação 5.2, onde Wh_Ah_B são os pesos entre as camadas A e B, *D* é a máscara *D* e Y é a saída da camada A. O operador .* significa uma multiplicação elemento a elemento e não uma multiplicação matricial normal.

$$net_h = (Wh_Ah_B .* D) * Y \tag{5.2}$$

Além da máscara *D*, há outra modificação proposta neste estudo que envolve a mudança da função de ativação dos neurônios da camada B. Em vez de usar a função sigmoide, a função identidade é adotada. Essa mudança faz com que a saída dos neurônios da camada B seja exatamente igual à entrada, que é a saída da camada A. Portanto, cada neurônio da camada B terá a mesma saída que o neurônio correspondente na camada A, garantindo uma equivalência direta entre as camadas. Isso facilita a preservação do aprendizado anterior e possibilita uma transição suave entre as camadas, contribuindo para um treinamento mais estável e eficiente da rede.

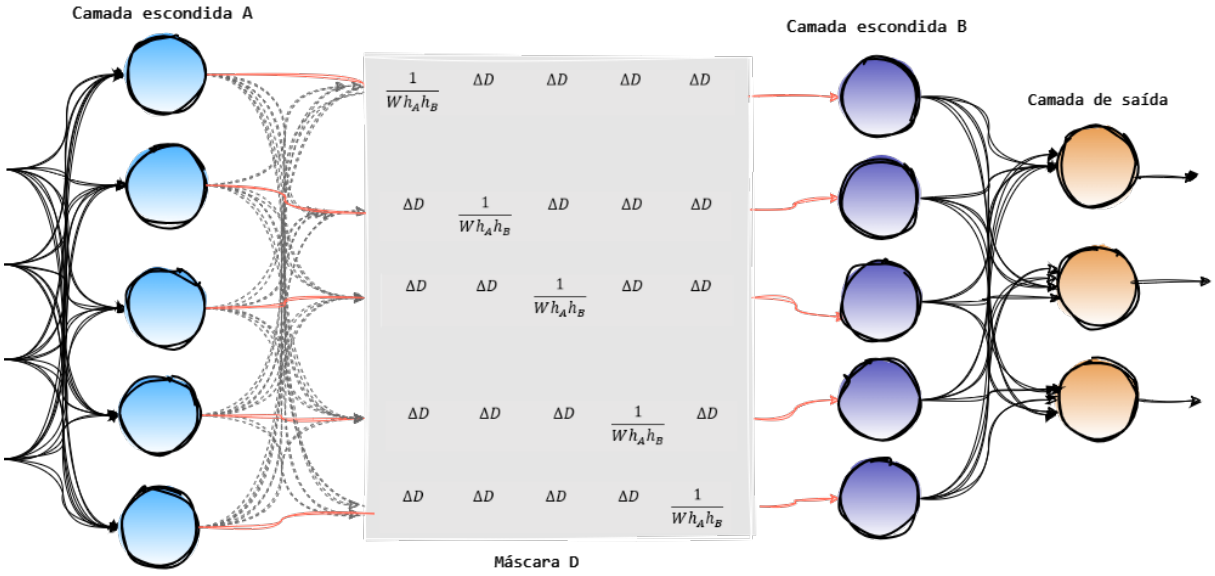
No entanto, as propostas de inovação apresentadas, na sua forma atual, não possibilitam que a nova camada tenha capacidade de aprendizado. Isso ocorre porque a saída da camada B sempre é semelhante à saída da camada A, resultando em nenhuma redução no erro de saída da rede através da modificação dos novos pesos criados. Portanto, após a inserção da camada B e a execução do algoritmo de treinamento da rede, tanto a máscara *D* quanto a função de ativação dos neurônios da camada B foram alteradas, a fim de permitir que exerçam influência sobre a saída da rede.

Entretanto, uma remoção indelicada da máscara *D* e/ou a troca da função de ativação, de identidade para sigmoide, resultam, de um modo, na capacidade de aprendizado da camada B, mas, de outro modo, ocasionam um aumento súbito no erro de saída da rede. Portanto, é essencial que ocorra uma transição suave e gradual ao retirar a máscara *D* e converter a função identidade para sigmoide.

Durante o treinamento, a transição referida no parágrafo anterior tem de acontecer. Após um determinado número de épocas, os pesos entre as camadas A e B devem ser ajustados

para reduzir o erro de saída da rede, sem prejudicar a queda do erro favorecido pelas camadas anteriores. O efeito desses novos pesos no treinamento é realizado mudando a máscara D em uma matriz com todos os elementos unitários. Essa mudança é efetuada através de um ΔD, o qual varia de 0 a 1, com velocidade estabelecida através da parametrização da inicialização da nova camada, como é possível observar na Figura 32. Dessa forma, o objetivo é garantir que a perturbação causada pela inserção de uma nova camada escondida na rede seja mínima.

Figura 32 – Alteração dos valores da máscara D por meio do ΔD.



Fonte: Elaborado pela autora (2024).

Na parametrização da inicialização da nova camada, é definido o valor do $salto_D$, que é a quantidade de épocas para incrementar o valores contidos na máscara D. Durante o tempo em que todos os elementos da máscara D variam uniformemente, os elementos da sua diagonal principal variam conforme os pesos aleatórios produzidos na inicialização da nova camada. Dependendo dos valores iniciais desses pesos, os valores da diagonal principal convergem mais rapidamente para 1.

No Algoritmo 1, é possível observar o pseudocódigo da CollabNet. O parâmetro que define a estabilidade do erro durante o treinamento, mencionado na linha 7, é a variação do erro, que exige um estudo específico para alcançar um desempenho adequado em relação à base de dados usada, à velocidade de alteração da variável c, ao número de neurônios nas camadas e a outros parâmetros relevantes. Nessa abordagem, a decisão de incluir uma nova camada é deixada a critério do projetista, que deve avaliar cuidadosamente as características do problema em questão.

Algoritmo 1 Pseudocódigo da CollabNet.

Entrada: Um conjunto de n objetos de treinamento.
Saida: CollabNet com pesos ajustados.

```
1: Inicio
2: Inicializar os pesos
3: while erro ≠ 0 do
4:     for i = 1,2,...n, para cada objeto x_i do conjunto de treinamento do
5:         Calcular valor de saída produzido pelo neurônio, f(x_i)
6:         Calcular erro = y_i − f(x_i)
7:         if erro > 0 and (erro não estabilizado) then
8:             Ajustar os pesos do neurônio
9:         else if erro > 0 then
10:            x = saída camada_{i−1}
11:            Incluir nova camada
12:        end if
13:    end for
14: end while
```

5.3.2 Variável c

Na CollabNet, a função de ativação dos neurônios da nova camada inserida também deve receber um tratamento. A ideia proposta pelo Lima Junior (2018) é que ao início do treinamento a função de ativação seja a função identidade, e com o passar do tempo, esta convirja para a função de ativação definida pelo projetista, podendo essa ser a função sigmoide, tangente hiperbólica, gaussiana, entre outras. Devido a isso ela é chamada de função de ativação mutante.

A ideia por trás da variável c é permitir, mesmo com a inclusão da máscara e alteração na função de ativação dos neurônios, que o aprendizado da rede ainda seja obtido por meio da variação dos pesos de forma gradual e sucessiva.

Além do próprio valor da variável c, a Collabnet ainda possui duas outras variáveis de controle, o Δc e o $salto_c$. Este último, refere-se a que passo das épocas pretende-se incrementar o valor da variável c e o primeiro o quanto deste valor é incrementado. A escolha desses hiper parâmetros é fundamental para o sucesso da técnica, pois quanto maior o valor da variável c, maior é a influência da nova camada no aprendizado, o que impacta diretamente no erro de saída da rede. Logo, direcionar o aprendizado da rede, tendo em vista a relação entre esses parâmetros e o aprendizado da rede, é a principal motivação na orientação dos experimentos apresentados neste trabalho.

Na Equação 5.3, foi determinada a função de ativação mutante da nova camada inserida (B), a qual seu valor depende de x e de c. O valor de x é o resultado da multiplicação da entrada pela matriz de pesos somado com o *bias*, ou simplesmente o net_h da Equação 5.2. O valor de c inicia-se igual a 0 e é incrementado até 1, conforme a evolução do treinamento. Essa função é dita "sigmoide mutante", pois quando o valor de c é 0, a função assume a mesma característica da função identidade $f(x) = x$, no entanto, ao passo que c tende a 1, a função comporta-se igual

a função sigmoide.

$$\phi_B(x,c) = \frac{1}{1+e^{-x}} \times c + x \times (1-c) \tag{5.3}$$

Consequentemente, a derivada da função de ativação deve ser alterada para o uso no *backpropagation*, como apresenta a Equação 5.4 em que y é a função sigmoide e c é o fator de ponderação.

$$\frac{d\phi_B}{dx} = c \times y \times (1-y) + (1-c) \tag{5.4}$$

Com a utilização da função de ativação mutante e da máscara D foi garantido que a inserção de uma nova camada não perturbou o erro da rede, e a medida que o treinamento continua, a nova camada se torna uma camada comum e devido ao treinamento e ao *backpropagation* o erro é diminuído. Após o erro estabilizar uma nova camada é inserida e os pesos das camadas existentes são congelados.

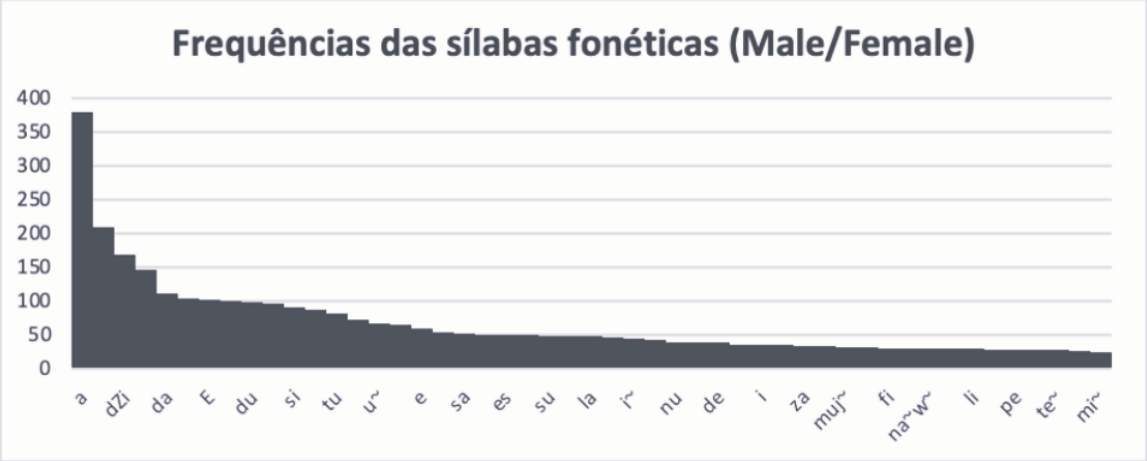
6 RESULTADOS E DISCUSSÃO

Neste capítulo, são apresentados os resultados obtidos durante este estudo. A Seção 6.1 apresenta o desbalanceamento da base de dados Male/Female. Na Seção 6.2, são apresentadas a definição do subconjunto utilizado, a extração dos centroides, a definição da quantidade de centroides utilizada e demais hiperparâmetros da CollabNet. A Seção 6.3 apresenta as métricas utilizadas para a avaliação dos resultados obtidos e a interpretação destes.

6.1 BASE DE DADOS

A base de dados Male/Female apresentou 447 classes de sílabas. Não obstante, não houve como trabalhar com todas pois a quantidade de amostras era desequilibrada. Por exemplo, existia muitas classes que possuíam apenas duas amostras, quase em torno de 60% do *dataset*. A Figura 33 exibe um subconjunto específico de sílabas fonéticas derivadas por meio dos modelos do FalaBrasil.

Figura 33 – Demonstração parcial das sílabas fonéticas extraídas dos modelos do FalaBrasil e classificadas por sua frequência de ocorrência no conjunto de dados Male/Female.



Fonte: Elaborado pela autora (2024).

6.2 CENTROIDES E HIPERPARÂMETROS

Devido ao desequilíbrio apresentado na Seção 6.1, era necessário o uso de técnicas de *data augmentation* a fim de melhorar a qualidade dos dados, para que dessa maneira o desempenho do modelo de aprendizado melhorasse. A ideia foi realizar o alinhamento forçado das sílabas fonéticas, igual a da Figura 20, com a base de dados LaPS Benchmark, e em seguida verificar a quantidade de amostras das classes que existiam nas duas bases. Por fim, fazer uma normalização entre as bases, por classes, de maneira que as amostras tivessem os mesmos limites de amplitudes. A Laps Bechmark é uma base 3.6 vezes maior que a Male/Female. A Male/Female tem 15 minutos de duração, enquanto a Laps Bechmark tem 54 minutos. Devido ao custo computacional, recursos (computador e tempo de pesquisa) e tempo de processamento

não foi possível seguir e fazer testes com a rede usando essa estratégia. Ela demorou muito mais tempo para ser alinhada.

Outra ideia para o Data Augmentation era criar amostras fictícias, ou seja, selecionar duas amostras e tirar a(s) média(s) do(s) centroide(s) dela(s), isso geraria um centroide fictício. Não se optou por isso inicialmente pois a ideia dessa pesquisa é criar um produto se aproxime da realidade e se tem disponível a Laps Benchmark para fazer os experimentos de maneira a encontrar esse produto. Devido ao tempo, não foi possível gerar os fictícios e fazer testes com eles.

Como observado na Figura 33, a sílaba *a* era a classe que apresentou mais amostras, um total de 379. E na mesma Figura, pode-se observar que o subconjunto dela tem 25 classes, onde 14 têm menos de 50 amostras. Após verificar as quantidade de amostras das 447 classes, foi decido utilizar um subconjunto da base Male/Female de 43 classes, sendo que cada classe apresentaria 30 amostras para que não ficasse desbalanceado. Na Tabela 3, tem-se o subconjunto da Male/Female que apresentou 30 ou mais amostras.

Tabela 3 – Subconjunto da base Male/Female que apresentou 30 ou mais amostras.

Classe	Quantidade total de amostras na base	Classe	Quantidade total de amostras na base	Classe	Quantidade total de amostras na base
a	379	*vi*	47	*ku*	30
dZi	168	*tSi*	103	*la*	48
da	112	*ta*	87	*li*	30
du	98	*tu*	82	*me∼*	35
E	102	*u*	146	*mi*	36
e	59	*u∼*	66	*mu*	30
ka	72	*de*	38	*muj∼*	32
ko∼	64	*es*	50	*na∼w∼*	30
ma	97	*e∼*	33	*nu*	39
na	54	*fi*	30	*pa*	50
ra	100	*i*	36	*sa∼w∼*	48
sa	52	*i∼*	44	*se*	50
si	90	*ki*	42	*su*	48
sil	210	*ko*	38	*te*	32
za	34				

Fonte: Elaborado pela autora (2024).

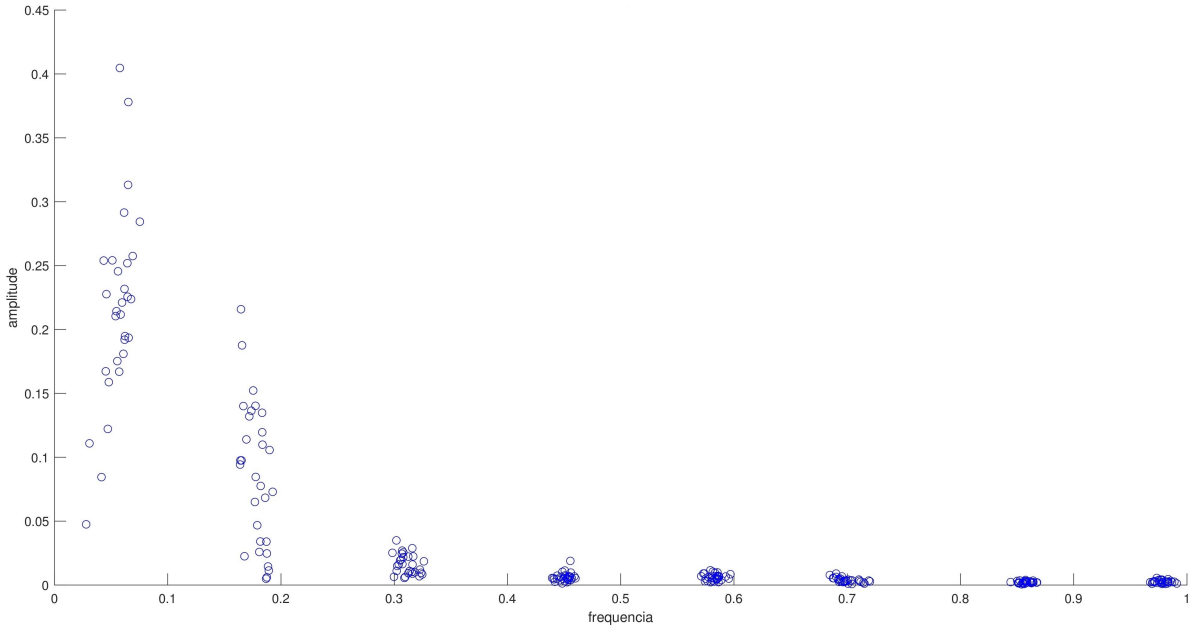
Ao trabalhar com as 43 classes, onde cada uma tem 30 amostras, tem-se um subconjunto com 1290 amostras ao total. Esse subconjunto foi dividido numa proporção de 80% para treinamento e 20% para teste, sendo então que, cada classe tinha 24 amostras para treino e 6 para teste na rede CollabNet.

Outro ponto a se definir era a quantidade de centroides que seriam extraídos dessas amostras para serem os dados de entrada da rede. A quantidade de centroides é uma variável importante para a representação da amostra. Isto impacta no aprendizado da rede. Ao mesmo

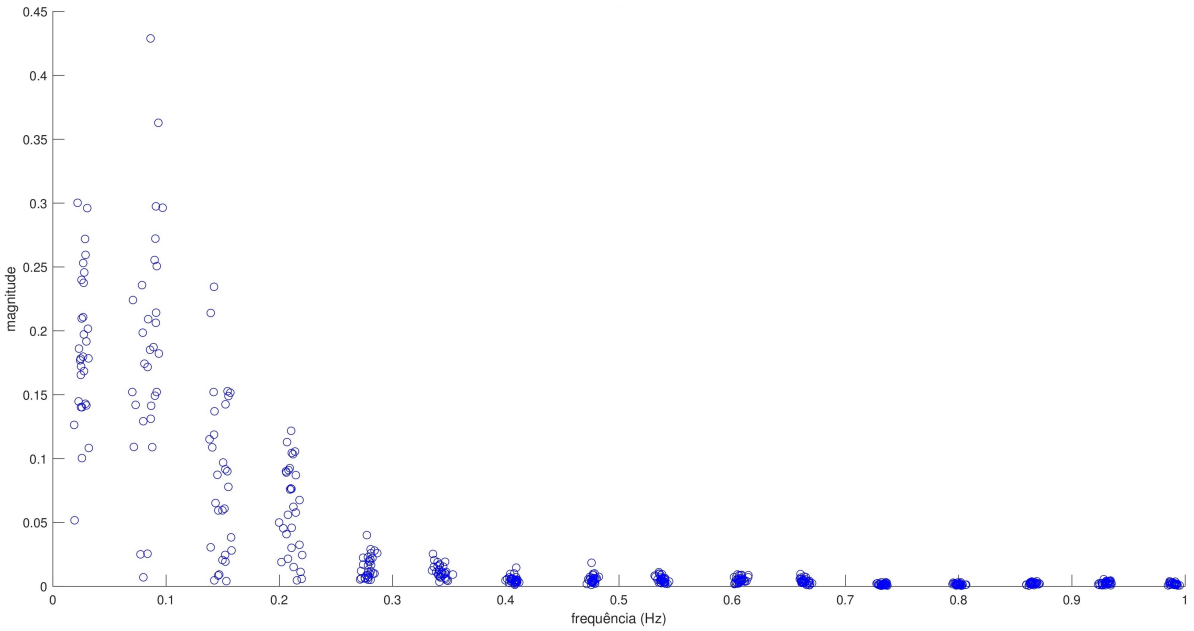
tempo que tem a finalidade de diminuir o custo computacional, eles precisam caracterizar a amostra.

Para isto, foram gerados os gráficos para visualização dos centroides das classes, como por exemplo, as Figuras 34 e 35 apresentam os gráficos de oito e 16 centroides das amostras do subconjunto da base Male/Female, das classes *ka* e *ku* respectivamente.

Figura 34 – Gráficos dos centroides das amostras do subconjunto da base de dados Male/Female da classe (sílaba fonética) *ka*.



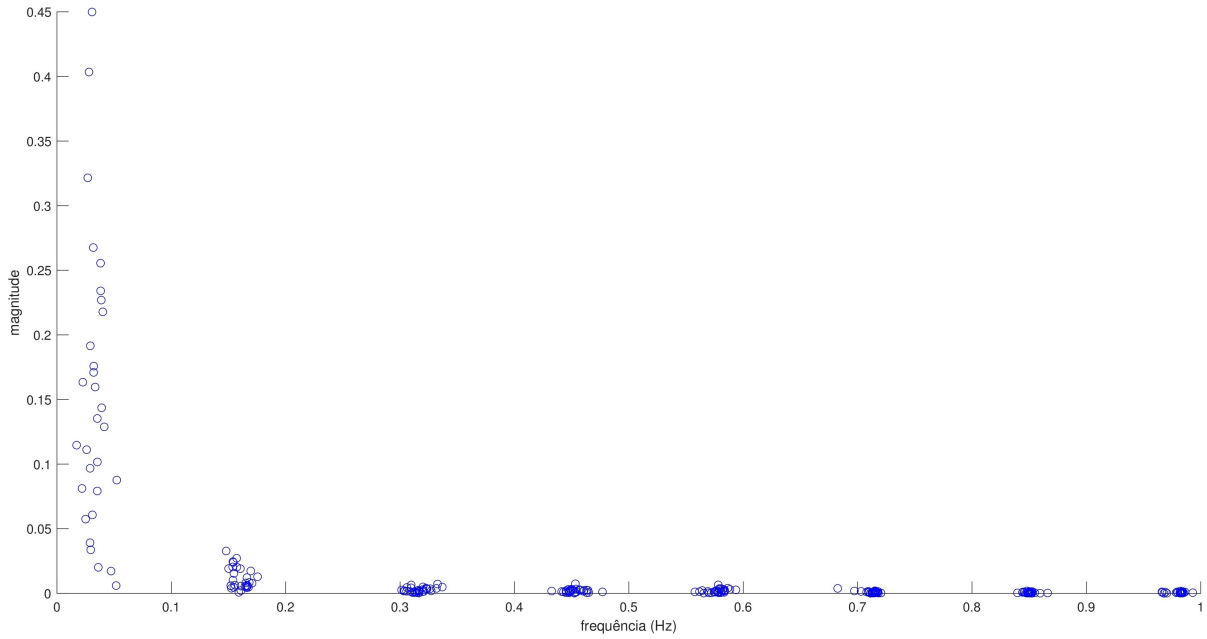
(a) Amostras da classe *ka* representadas por oito centroides característicos.



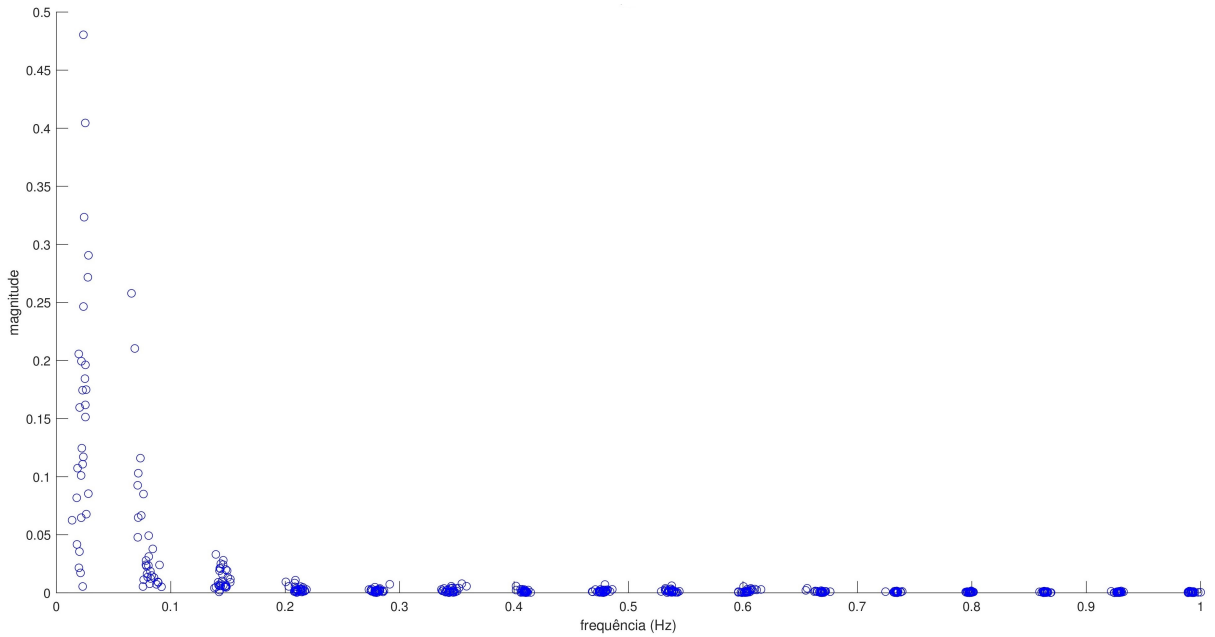
(b) Amostras da classe *ka* representadas por 16 centroides característicos.

Fonte: Elaborado pela autora (2024).

Figura 35 – Gráficos dos centroides de todas as amostras existentes na base de dados Male/Female da classe (sílaba fonética) *ku*.



(a) Amostras da classe *ku* representadas por oito centroides característicos.



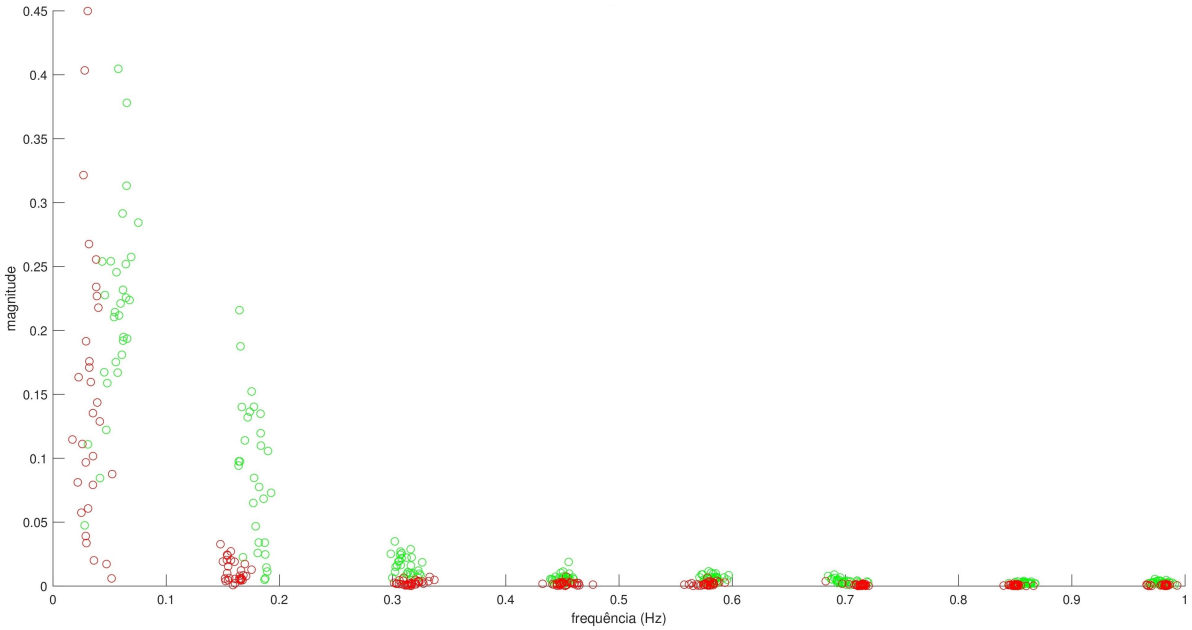
(b) Amostras da classe *ku* representadas por 16 centroides característicos.

Fonte: Elaborado pela autora (2024).

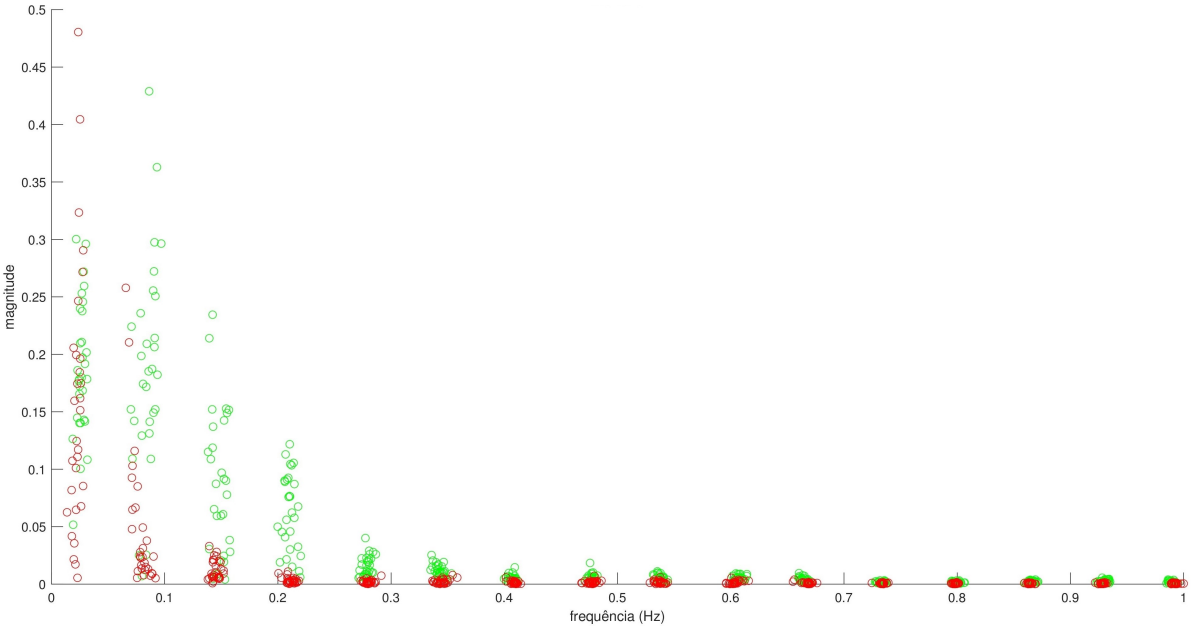
As Figuras 34 e 35 apresentam 30 centroides em cada grupo de frequências, onde cada centroide é respectivo a uma amostra do subconjunto da base Male/Female. Por exemplo, na Figura 34a, têm-se oito grupos de frequências, onde em cada grupo, 30 centroides foram desenhados, cada um de uma respectiva amostra da classe *ka*. Em seguida, ainda como exemplo

das classes *ka* e *ku*, foram desenhados os centroides de todas as amostras do subconjunto da base Male/Female, comparando essas classes.

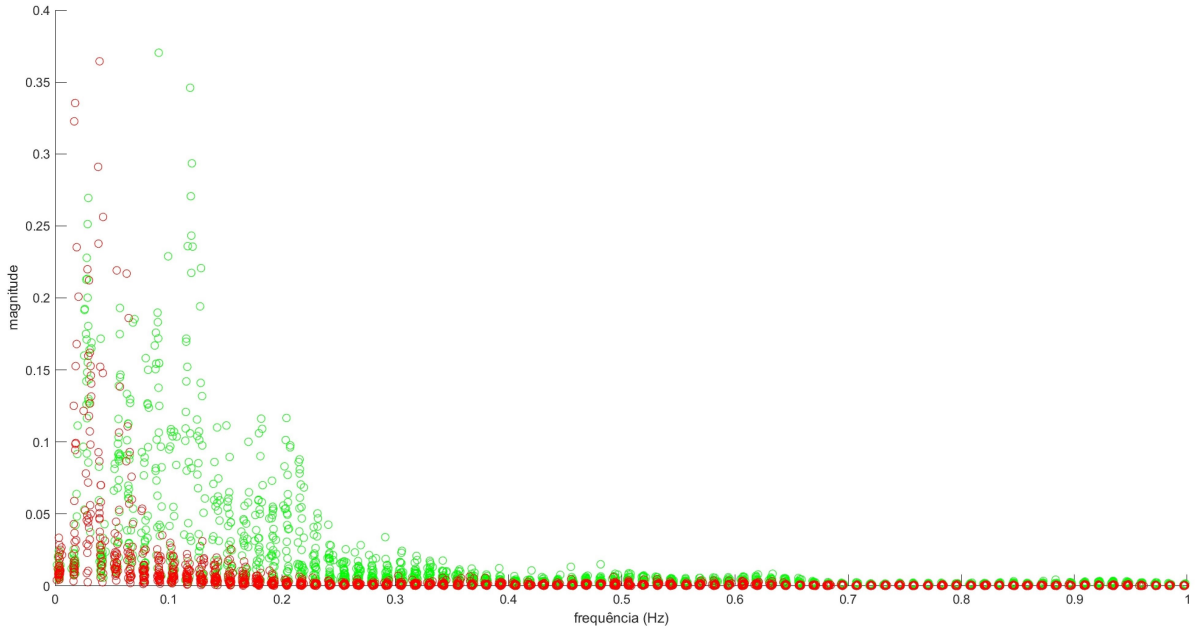
Figura 36 – Gráficos dos centroides das amostras do subconjunto da base Male/Female das classes (sílabas fonéticas) *ka* e *ku* nas cores verde e vermelho, respectivamente.



(a) Amostras das classes *ka*, em verde, e *ku*, em vermelho, representadas por oito centroides característicos.



(b) Amostras das classes *ka*, em verde, e *ku*, em vermelho, representadas por 16 centroides característicos.

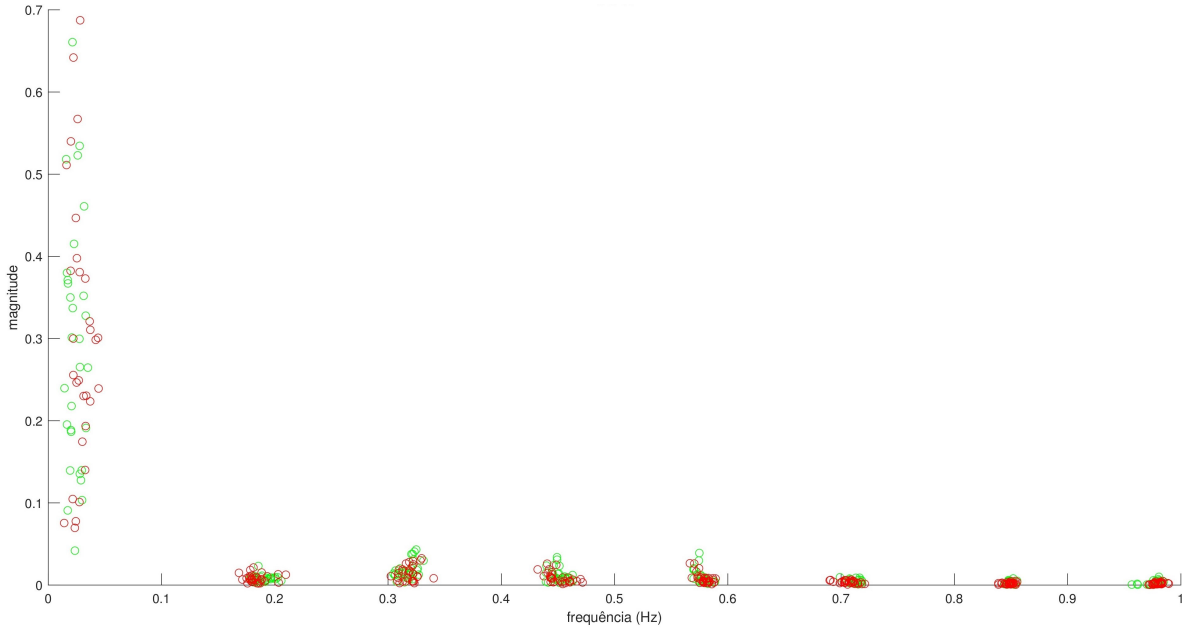


(c) Amostras das classes *ka*, em verde, e *ku*, em vermelho, representadas por 80 centroides característicos.

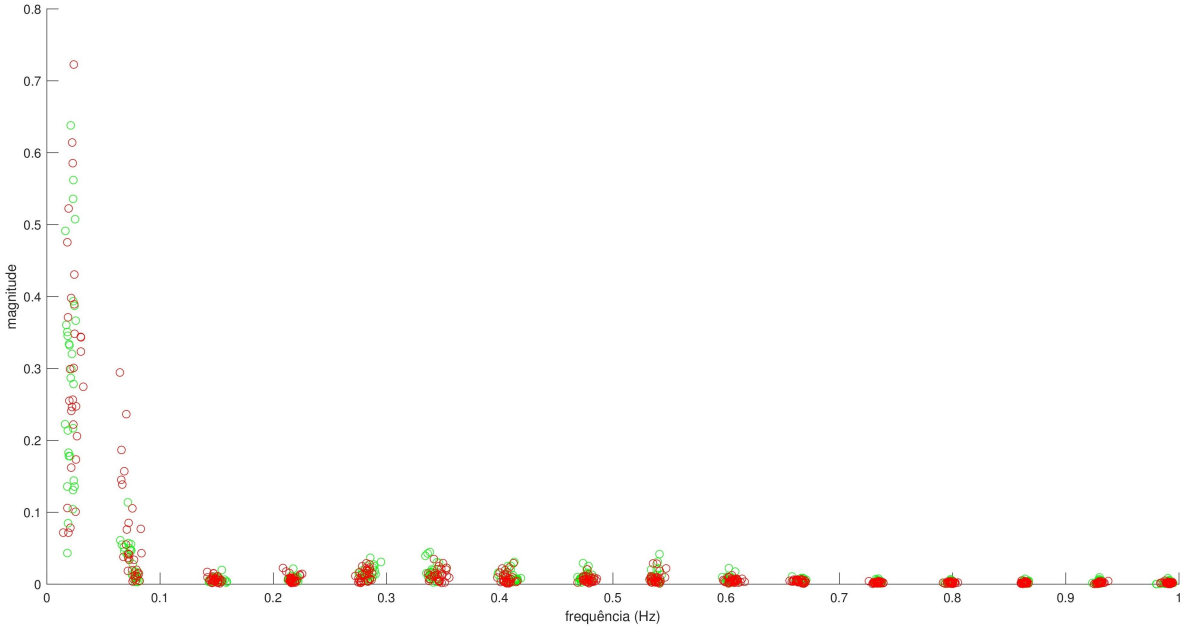
Fonte: Elaborado pela autora (2024).

Foram comparadas classes que contêm um som surdo e outro sonoro, conceito explanado na Seção B.1 do Apêndice B. Como exemplo, as classes *vi* e *da* são sonoras, já as classes *fi* e *ta* são surdas. As Figuras 37, 38, 39 e 40 apresentam os gráficos dessas comparações.

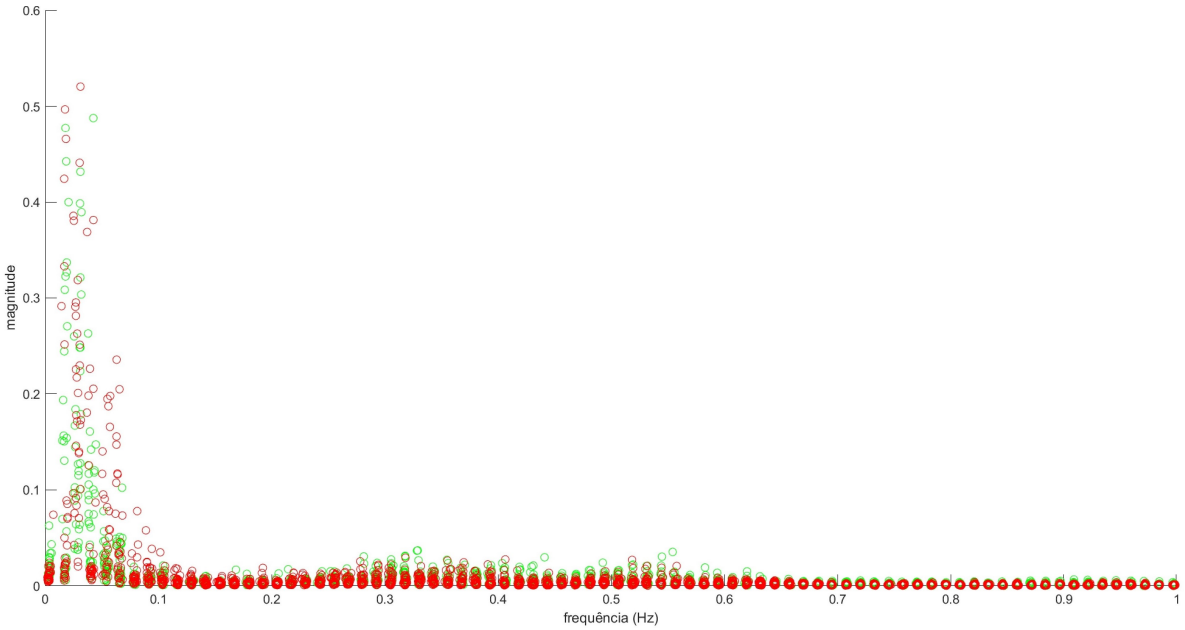
Figura 37 – Gráficos dos centroides das amostras do subconjunto da base Male/Female das classes (sílabas fonéticas) *fi* e *vi* nas cores verde e vermelho, respectivamente.



(a) Amostras das classes *fi*, em verde, e *vi*, em vermelho, representadas por oito centroides característicos.



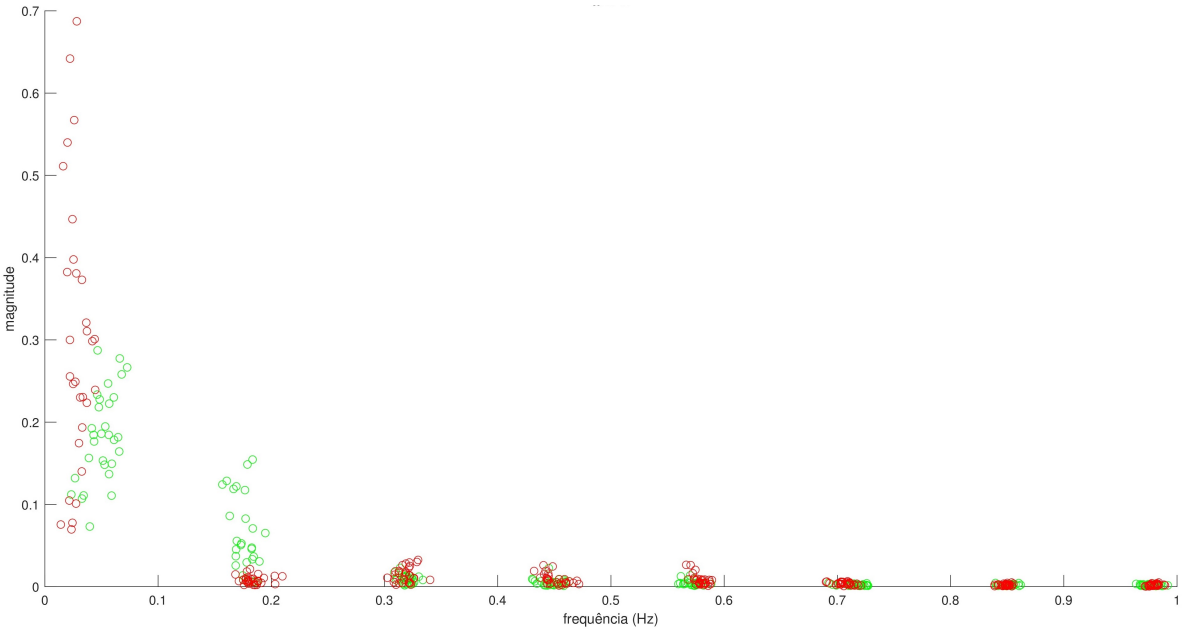
(b) Amostras das classes fi, em verde, e vi, em vermelho, representadas por 16 centroides característicos.



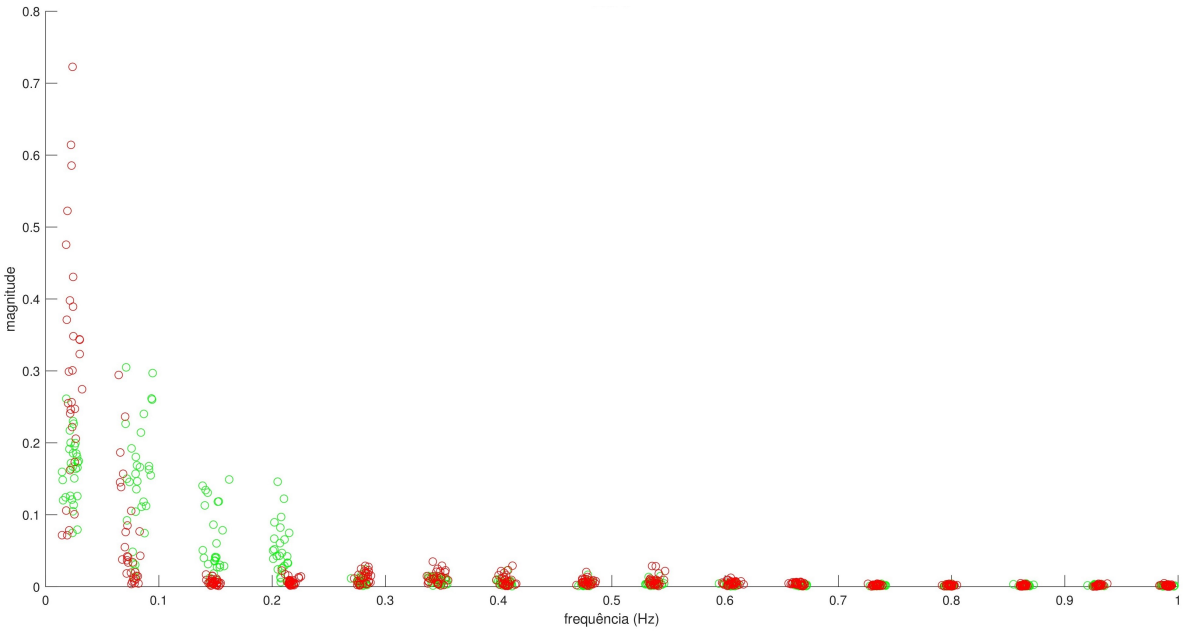
(c) Amostras das classes fi, em verde, e vi, em vermelho, representadas por 80 centroides característicos.

Fonte: Elaborado pela autora (2024).

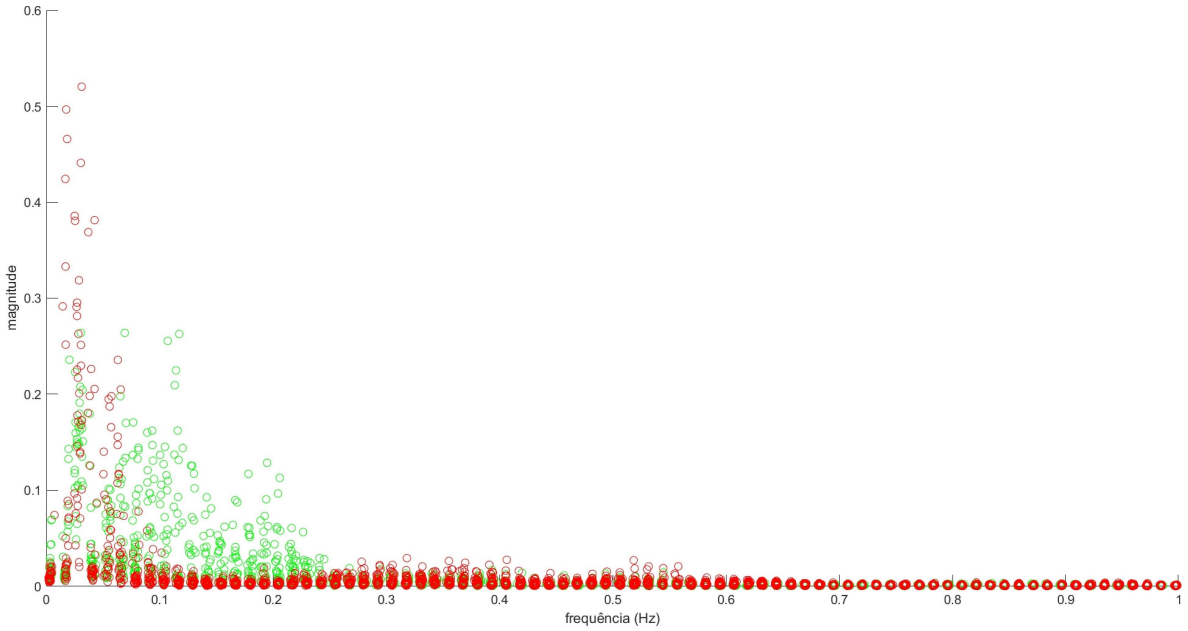
Figura 38 – Gráficos dos centroides das amostras do subconjunto da base Male/Female das classes (sílabas fonéticas) *ta* e *vi* nas cores verde e vermelho, respectivamente.



(a) Amostras das classes *ta*, em verde, e *vi*, em vermelho, representadas por oito centroides característicos.



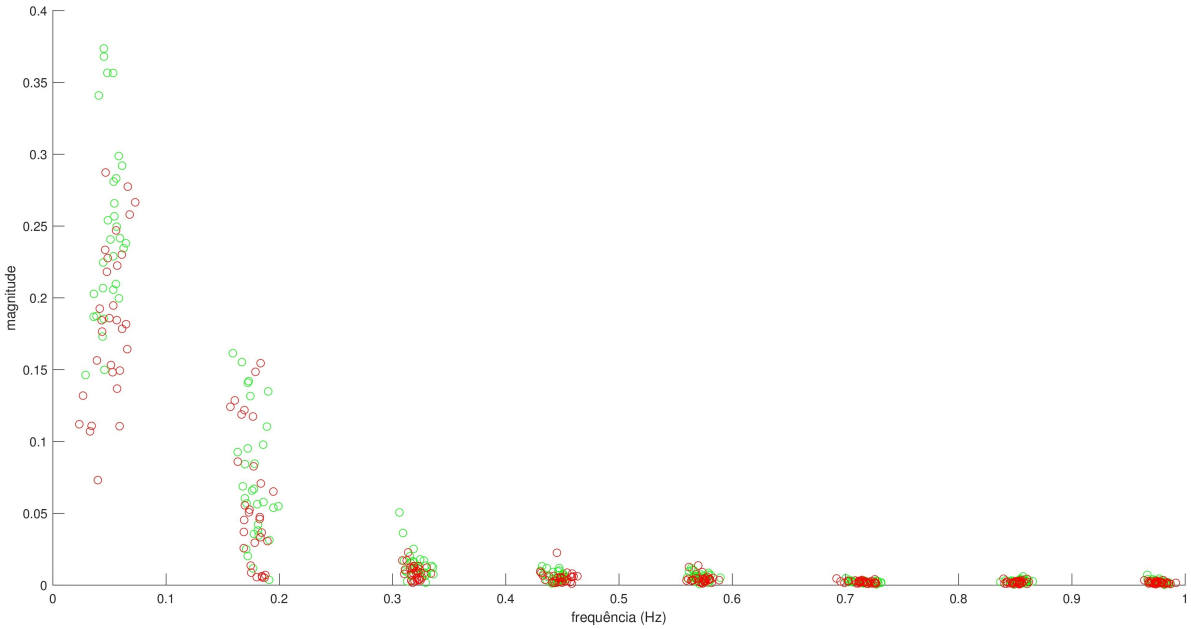
(b) Amostras das classes *ta*, em verde, e *vi*, em vermelho, representadas por 16 centroides característicos.



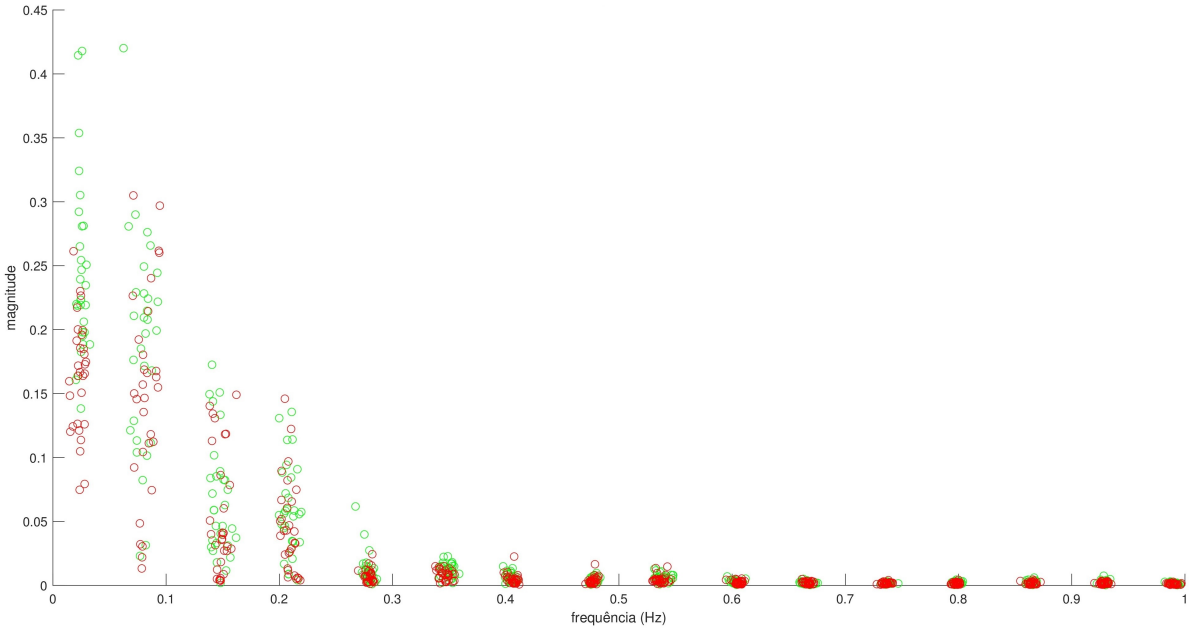
(c) Amostras das classes *ta*, em verde, e *vi*, em vermelho, representadas por 80 centroides característicos.

Fonte: Elaborado pela autora (2024).

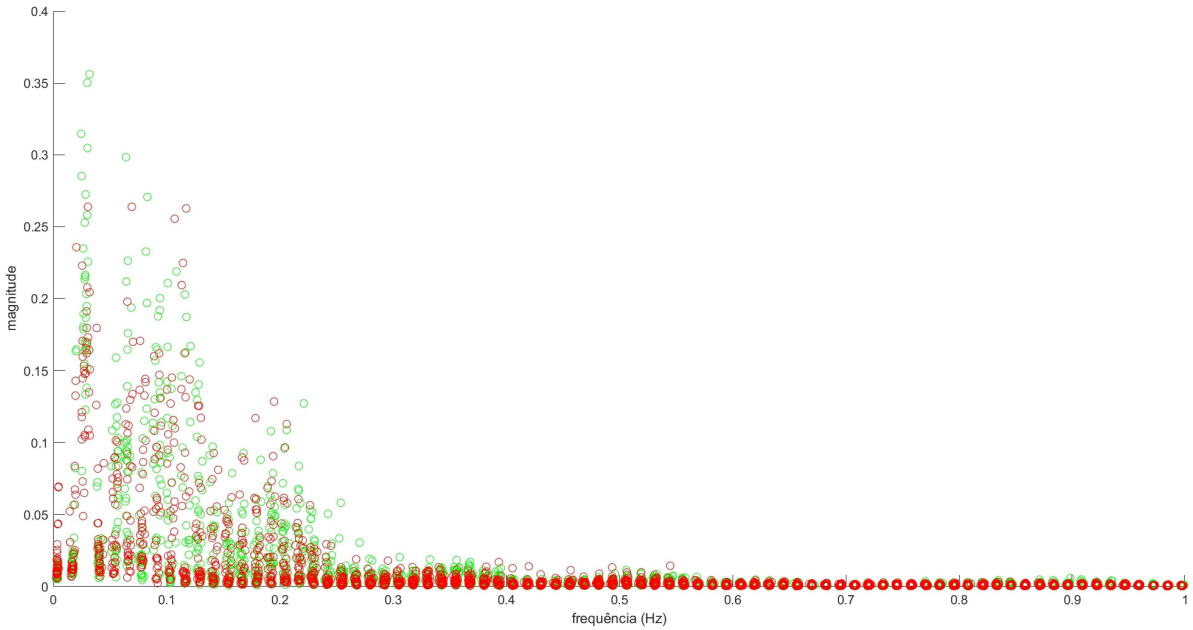
Figura 39 – Gráficos dos centroides das amostras do subconjunto da base Male/Female das classes (sílabas fonéticas) *da* e *ta* nas cores verde e vermelho, respectivamente.



(a) Amostras das classes *da*, em verde, e *ta*, em vermelho, representadas por oito centroides característicos.



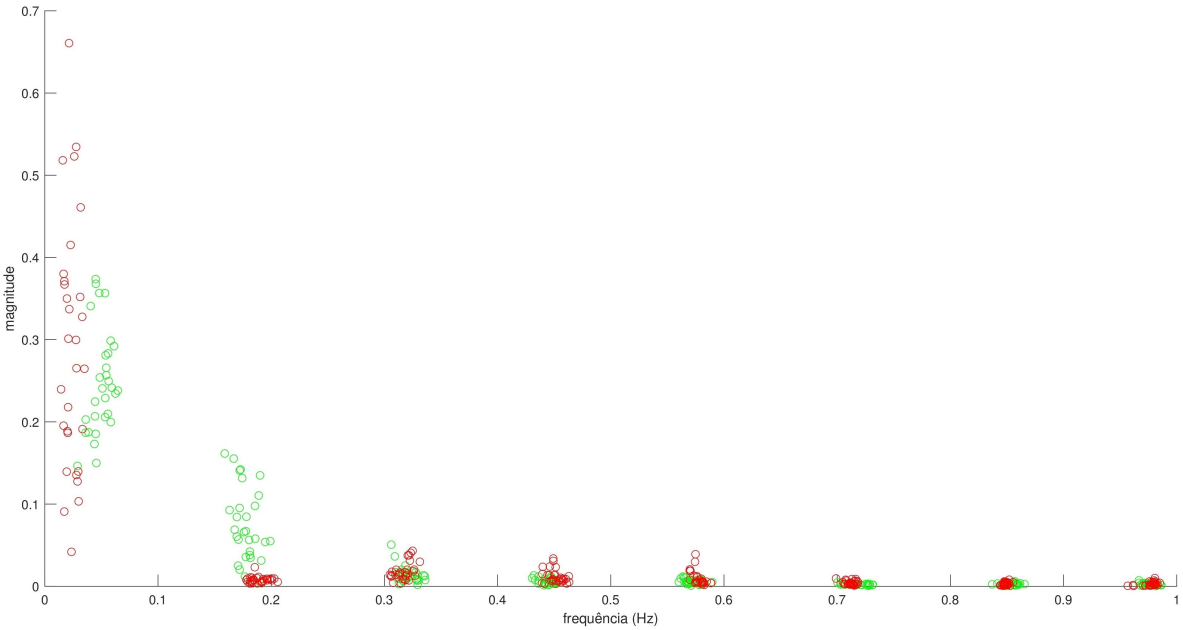
(b) Amostras das classes *da*, em verde, e *ta*, em vermelho, representadas por 16 centroides característicos.



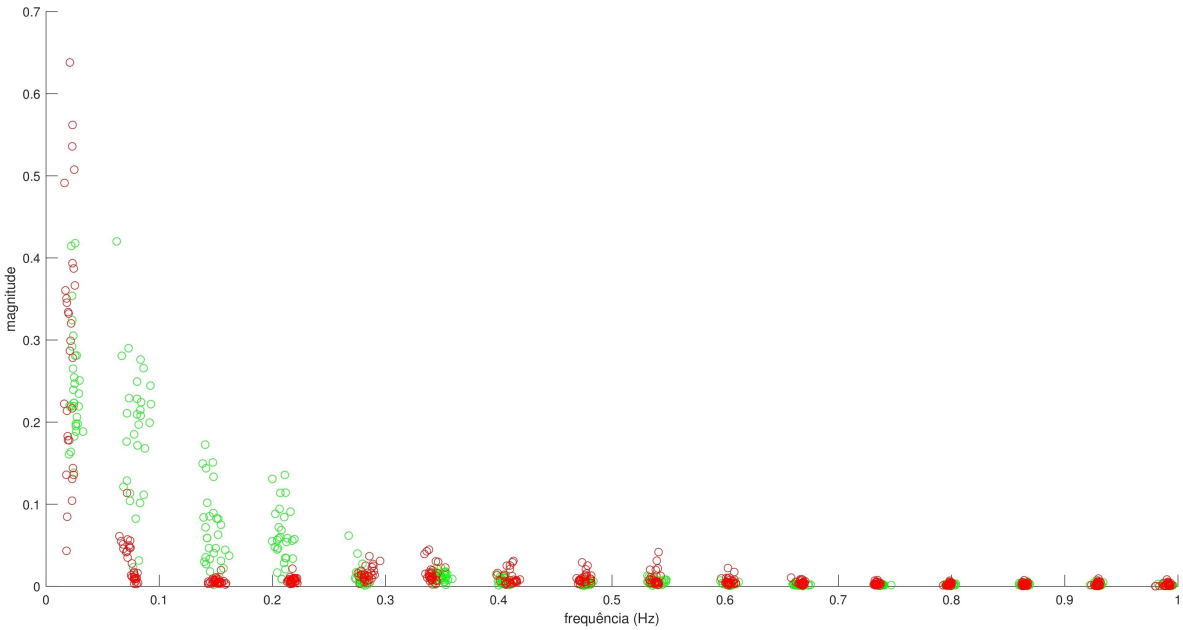
(c) Amostras das classes *da*, em verde, e *ta*, em vermelho, representadas por 80 centroides característicos.

Fonte: Elaborado pela autora (2024).

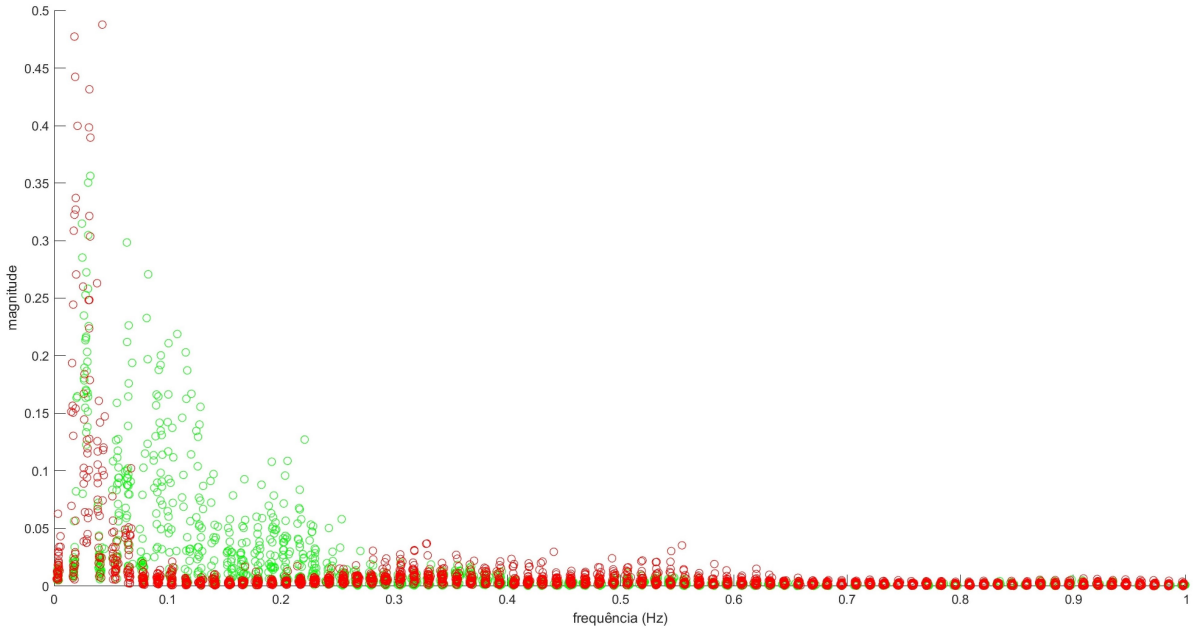
Figura 40 – Gráficos dos centroides das amostras do subconjunto da base Male/Female das classes (sílabas fonéticas) *da* e fi nas cores verde e vermelho, respectivamente.



(a) Amostras das classes *da*, em verde, e fi, em vermelho, representadas por oito centroides característicos.



(b) Amostras das classes *da*, em verde, e fi, em vermelho, representadas por 16 centroides característicos.



(c) Amostras das classes *da*, em verde, e fi, em vermelho, representadas por 80 centroides característicos.

Fonte: Elaborado pela autora (2024).

Pode-se observar nas Figuras 37 e 39, que sons parecidos têm centroides parecidos, em consequência dos gráficos de barras verticais serem parecidos. As Figuras 36, 38 e 40 apresentam os centroides de sons diferentes, por isso apresentam centroides diferentes em determinados grupos de barras verticais. Assim comprovando que sons diferentes apresentam centroides distintos e sons semelhantes apresentam centroides que se assemelham.

Após as análises dos gráficos, foi escolhida a quantia de 16 centroides para ser a representação das amostras. Logo, a camada de entrada da CollabNet possui 32 neurônios. Quanto à parametrização, a busca dos hiperparâmetros dos modelos foi conduzida empiricamente. É importante notar que, embora nenhuma técnica de otimização tenha sido empregada nesse estágio, os experimentos foram minuciosos para identificar as configurações de aprendizagem mais eficazes para o modelo, resultando em melhores desempenhos.

Assim, foram exploradas várias combinações de parâmetros, abrangendo não apenas os elementos que compõem a estrutura da rede neural, como o número de neurônios e camadas, mas também os relacionados à CollabNet: Δc, como o incremento da variável C; $salto_c$, representando o passo da iteração para atualização do C; $salto_D$, representando o passo da iteração para atualização da máscara D; e o valor inicial de C. O espaço de busca de hiperparâmetros utilizado nos experimentos é detalhado na Tabela 4.

Os hiperparâmetros tradicionais como taxa de aprendizado, número de épocas de treinamento de cada camada, quantidade de neurônios em cada camada, função de ativação e número de camadas ocultas, junto com os demais, foram escolhidos por meio de sucessivos experimentos.

Tabela 4 – Hiperparâmetros da CollabNet.

Hiper parâmetro	Domínio
Δc	[0, 1]
$salto_c$	[25, ε]
$salto_D$	[25, ε]
c	[0, 1]
Taxa de aprendizado	$[10^{-3}, 10^{-5}]$
Número de camadas	[1, 4]
Número de neurônios	[16, 30]

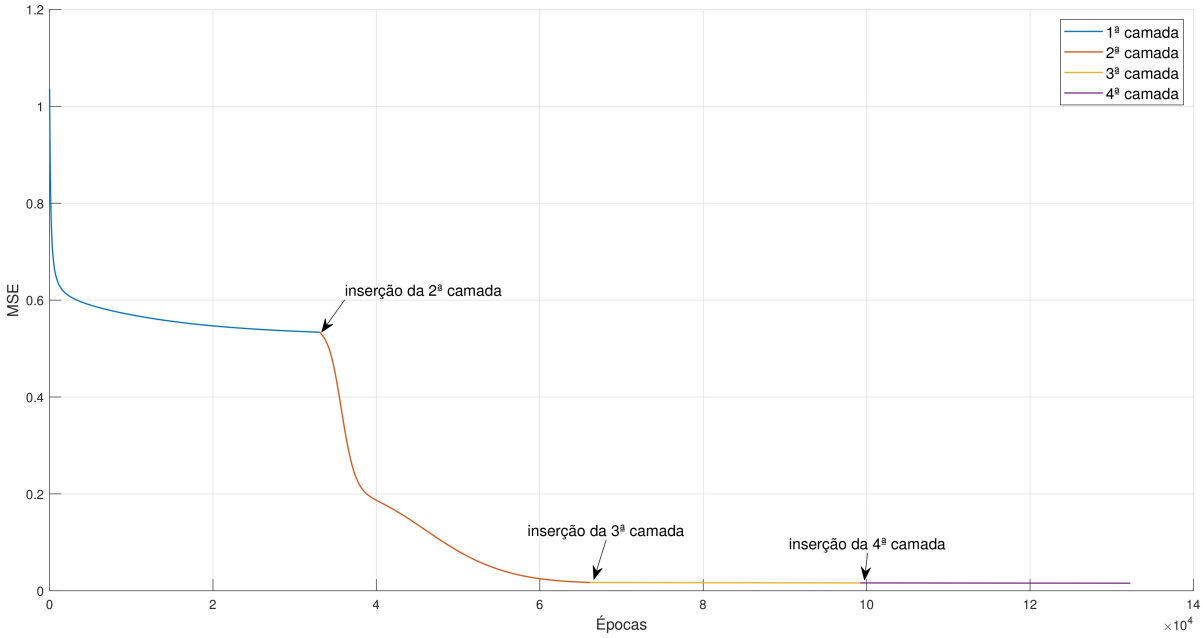
Fonte: Elaborado pela autora (2024).

Ao escolher os hiperparâmetros de forma precisa, o objetivo é integrar novas camadas em uma rede neural sem causar perturbações significativas no erro da rede e garantir a incorporação do aprendizado da nova camada. Na Figura 41, tem-se o gráfico do MSE (erro quadrático médio), definido na Equação 6.1, onde se pode observar as inserções das camadas ao decorrer do treinamento da CollabNet.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2 \tag{6.1}$$

onde n é a quantidade de amostras de treinamento, Y_i o valor real da classe e $\hat{Y}_i$ o valor predito da classe.

Figura 41 – Inclusão de cada camada durante o treinamento.

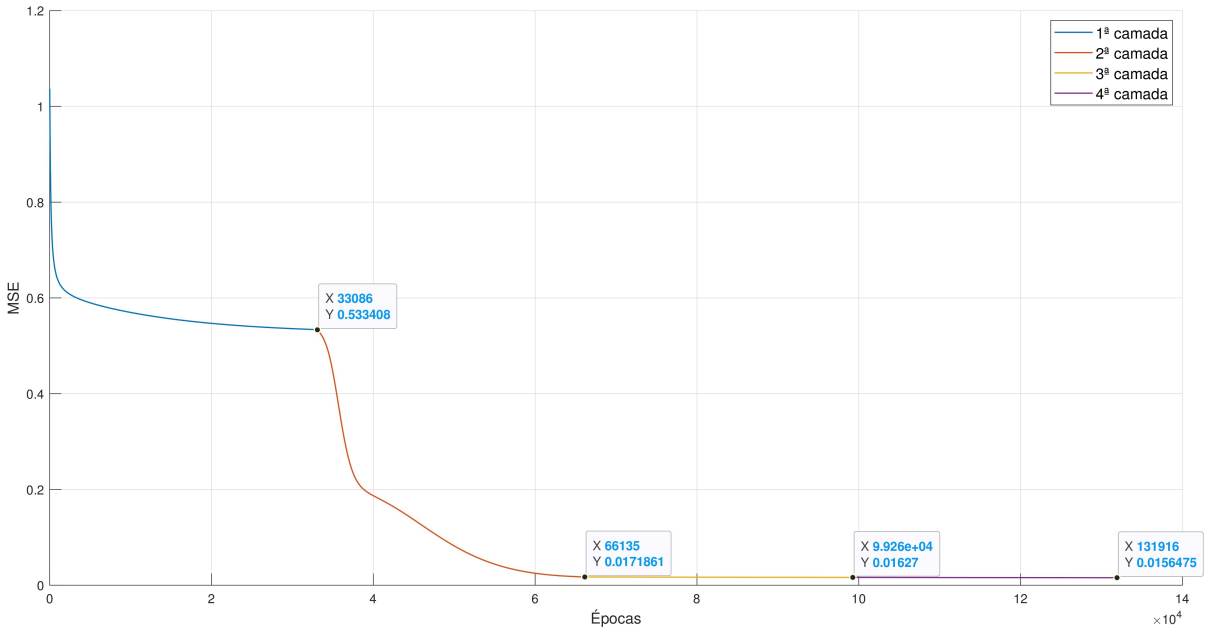


Fonte: Elaborado pela autora (2024).

O MSE final, após finalizado o treinamento em 132.269 épocas, foi de 0,0156, como observado na Figura 42. Essa figura apresenta os valores finais de MSE antes da inclusão de uma

nova camada. Antes de ser inserida a segunda camada, o MSE era 0,5339. O valor do MSE era 0,0171 antes de incluir a terceira camada. Por fim, antes da inserção da quarta camada, o valor do MSE era 0,0162.

Figura 42 – MSE a cada inclusão de uma nova camada.



Fonte: Elaborado pela autora (2024).

Pode-se observar que a cada inclusão durante o treinamento, o MSE diminui. Para validar o modelo, propõe-se a comparação dos resultados de teste com outros modelos da literatura. As métricas de avaliação serão detalhadas na próxima seção.

6.3 AVALIAÇÃO E INTERPRETAÇÃO

Para avaliar os resultados dos modelos, propõe-se a utilização das métricas de acurácia, sensibilidade, precisão e f-score. Além disso, é necessário um método que possibilite a correta classificação das sílabas fonéticas, ao mesmo tempo em que se verifica a não perturbação do erro global e a incorporação do aprendizado de cada nova camada. Para isso, é conduzida uma comparação do erro global e das métricas mencionadas acima ao término do treinamento de cada camada.

Para o cálculo das métricas acima citadas, são utilizadas as medidas FP (falso positivo), FN (falso negativo), VN (verdadeiro negativo) e VP (verdadeiro positivo), definidas da seguinte forma:

- FP: tem-se falso positivo para uma classe $'x'$, quando uma amostra de outra classe é reconhecida como da classe $'x'$.
- FN: tem-se falso negativo para a classe $'x'$, quando uma amostra da classe $'x'$ é reconhecida como de outra classe.

- VN: tem-se verdadeiro negativo para a classe $'x'$, quando uma amostra de outra classe é reconhecida como da outra classe.
- VP: tem-se verdadeiro positivo para a classe $'x'$, quando uma amostra da classe $'x'$ é reconhecida como da classe $'x'$.

Com essas medidas, pode-se calcular as métricas que são responsáveis pela avaliação dos resultados obtidos com o modelo. A acurácia, comumente reconhecida por oferecer uma avaliação geral do experimento, é definida como a proporção de todos os resultados corretos em relação à população do experimento (FAWCETT, 2006). Logo, o cálculo da acurácia (ACU) pode ser definido conforme a Equação 6.2:

$$ACU = \frac{VP+VN}{VP+VN+FP+FN} \tag{6.2}$$

A precisão (PRE), ou valor preditivo positivo, pode ser considerada como a taxa de acerto do modelo em relação as classificações das sílabas fonéticas. A PRE é dada pela Equação 6.3

$$PRE = \frac{VP}{VP+FP} \tag{6.3}$$

A sensibilidade (SEN) ou *recall*, tendo em vista todos os casos de classe positiva, é a taxa de acerto dos casos verdadeiros positivos. Portanto, a SEN é dada pela Equação 6.4:

$$SEN = \frac{VP}{VP+FN} \tag{6.4}$$

O f-score (FSC) considera tanto o cálculo da precisão quanto o da sensibilidade, sendo composto pela média harmônica de ambas as métricas. Ele é definido pela Equação 6.5

$$FSC = 2 * \frac{PRE * SEN}{PRE + SEN} \tag{6.5}$$

Por fim, a ultima métrica a ser utilizada para avaliar o desempenho de sistemas de reconhecimento de fala ou de tradução, denominada de *Phone Error Rate* (PER). A métrica PER leva em conta quatro parâmetros cruciais para avaliar o desempenho do sistema de reconhecimento: substituições (S), exclusões (D) e inserções (I), divididas pelo número total de predições (N), conforme mostrado na Equação 6.6.

$$PER = \frac{S+D+I}{N} \tag{6.6}$$

Para ilustrar o cálculo de exclusões, inserções e substituições na métrica PER, usam-se a seguir as sílabas si, sil e e dZi como exemplos. Para a sílaba si, houveram amostras identificadas como sil, ou seja, houveram inserções da letra “l”. Para a sílaba sil, houveram amostras identificadas como si, ou seja, houveram exclusões da letra “l”. Para a sílaba dZi, houveram uma substituição (sílaba tSi) e uma exclusão (sílaba i).

No total, após a inserção da quarta camada, para o conjunto de teste, o *PER* foi de 23,73%. A precisão foi de 81,77%, o *recall* foi de 75,96% e o f-score foi de 77,37%. Na Tabela 5, tem-se

o relatório de classificação das sílabas fonéticas, onde se apresenta as métricas mencionadas por classe e para o conjunto total de teste.

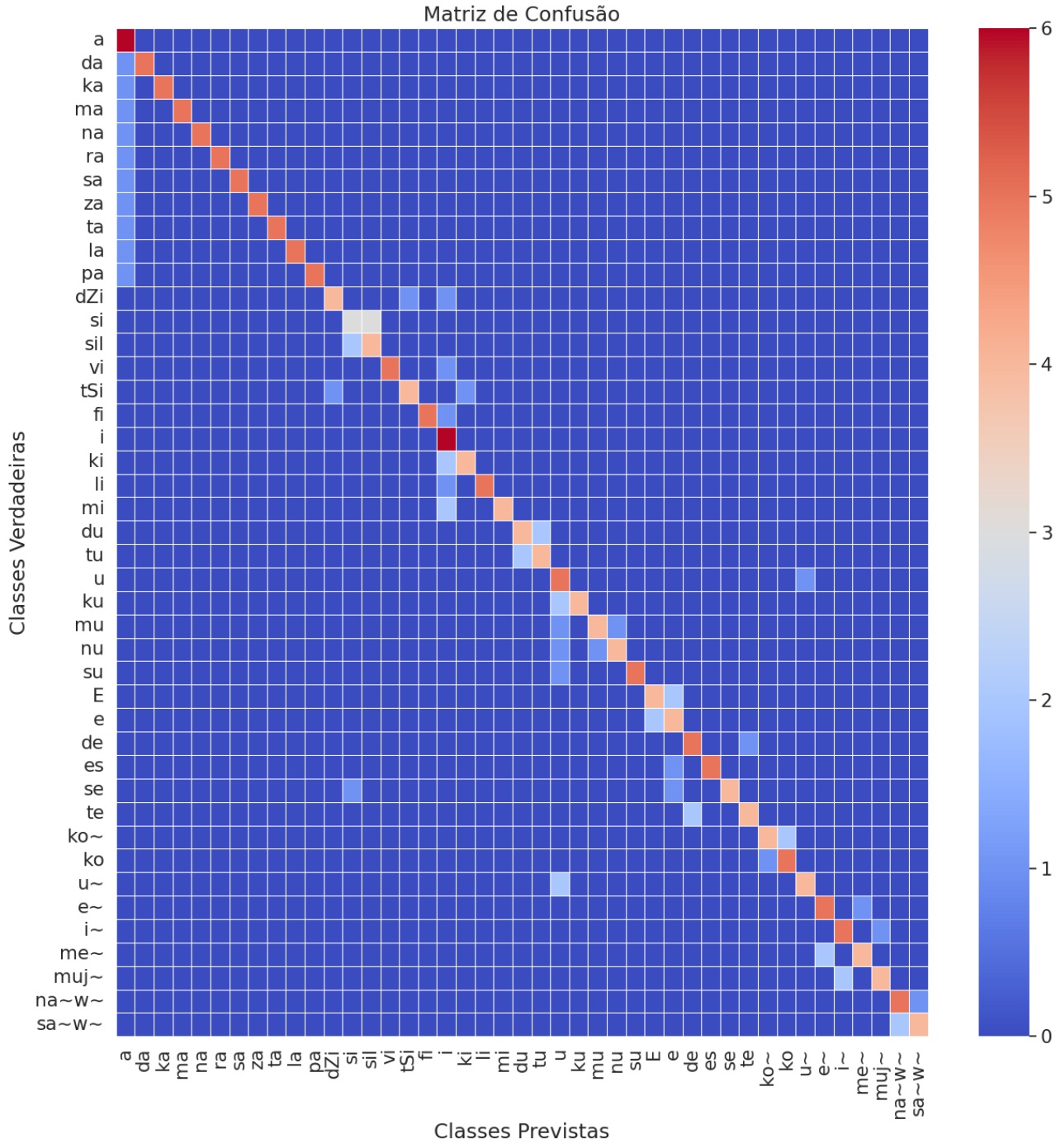
Tabela 5 – Relatório de classificação das sílabas fonéticas.

Relatório de classificação									
	PRE	SEN	FSC	suporte		PRE	SEN	FSC	suporte
a	0.38	1.00	0.55	6	mi	1.00	0.67	0.80	6
da	1.00	0.83	0.91	6	du	0.67	0.67	0.67	6
ka	1.00	0.83	0.91	6	tu	0.67	0.67	0.67	6
ma	1.00	0.83	0.91	6	u	0.42	0.83	0.56	6
na	1.00	0.83	0.91	6	ku	1.00	0.67	0.80	6
ra	1.00	0.83	0.91	6	mu	0.80	0.67	0.73	6
sa	1.00	0.83	0.91	6	nu	0.80	0.67	0.73	6
za	1.00	0.83	0.91	6	su	1.00	0.83	0.91	6
ta	1.00	0.83	0.91	6	E	0.67	0.67	0.67	6
la	1.00	0.83	0.91	6	e	0.50	0.67	0.57	6
pa	1.00	0.83	0.91	6	de	0.71	0.83	0.77	6
dZi	0.80	0.67	0.73	6	es	1.00	0.83	0.91	6
si	0.50	0.50	0.50	6	se	1.00	0.67	0.80	6
sil	0.57	0.67	0.62	6	te	0.80	0.67	0.73	6
vi	1.00	0.83	0.91	6	ko∼	0.80	0.67	0.73	6
tSi	0.80	0.67	0.73	6	ko	0.71	0.83	0.77	6
fi	1.00	0.83	0.91	6	u∼	0.80	0.67	0.73	6
i	0.43	1.00	0.60	6	e∼	0.71	0.83	0.77	6
ki	0.80	0.67	0.73	6	i∼	0.71	0.83	0.77	6
li	1.00	0.83	0.91	6	me∼	0.80	0.67	0.73	6
sa∼w∼	0.80	0.67	0.73	6	muj∼	0.80	0.67	0.73	6
					na∼w∼	0.71	0.83	0.77	6
ACU			0.76	258					
macro avg	0.82	0.76	0.77	258					
weighted avg	0.82	0.76	0.77	258					

Fonte: Elaborado pela autora (2024).

Na Figura 43, é apresentada a matriz de confusão do melhor modelo obtido ao longo dos experimentos. É possível verificar que a maioria das predições está de acordo com as classes correspondentes. A acurácia obtida, como já apresentada na Tabela 5, foi de 75,96%.

Figura 43 – Matriz de confusão.



Fonte: Elaborado pela autora (2024).

Na Tabela 6, são apresentados os trabalhos que utilizam bases de áudio da língua portuguesa brasileira para reconhecimento de fonemas. São apresentados a acurácia, o PER e o sistema de classificação dos mesmos.

Tabela 6 – Comparação com os trabalhos presentes na literatura.

Trabalhos da literatura	Sistema	PER (%)	ACU (%)
Nedjah *et al.* (2023)	CHEM-only	33,42	78,9
Nedjah *et al.* (2023)	CHEM+	13,13	86,53
Nedjah *et al.* (2023)	PEM-only	54,52	63,07
Nedjah *et al.* (2023)	PEM+	21,36	77,75
Catete e Rosa (2023)	HTK (HMM)	-	39,18
Quintanilha *et al.* (2017)	CNN+RNNs	25,13	-
Este trabalho	CollabNet	23,73	75,96

Fonte: Elaborado pela autora (2024).

Essa comparação é meramente realizada para se ter uma percepção frente a outros trabalhos de reconhecimento de fonemas da língua portuguesa. Como se pode observar, a CollabNet obteve um PER inferior a dois terços dos demais classificadores. A acurácia também manteve-se dentro do escopo dos trabalhos apresentados. Quando comparada aos classificadores que não utilizam mais de uma técnica, ele cumpre um desempenho superior ou semelhante, sendo a maior diferença de 2,94% de ACU. Já para os classificadores que utilizam técnicas combinadas, seu PER é maior, mas a ACU não se mantém tão distante, sendo a maior diferença de 10,57%.

7 CONCLUSÃO

Este trabalho apresentou uma rede *deep stacked autoencoder*, denominada CollabNet, que reconhece sílabas fonéticas representadas por meio de grupos de centroides, após aplicada a FFT. O reconhecimento de sílabas fonéticas é uma etapa essencial em muitos sistemas de processamento de fala. Esses sistemas desempenham um papel crucial em uma variedade de aplicações, desde assistentes de voz em dispositivos móveis até sistemas de reconhecimento de fala em carros e em ambientes domésticos inteligentes.

O processo de reconhecimento de sílabas fonéticas ocorre em várias etapas. Primeiro, o sinal de fala é pré-processado para extrair características relevantes. Neste trabalho, para a primeira etapa, foi desenvolvido um coeficiente novo obtido a partir de centroides do gráfico de barras das frequências. Esse método é interessante pois ele foi capaz de reduzir a quantidade de dados necessária para fazer o reconhecimento das sílabas fonéticas, sem perder as características próprias que o sinal sonoro carrega.

Na segunda etapa, essas características são usadas para modelar as unidades fonéticas, que podem incluir fonemas, sílabas ou até mesmo unidades subfonêmicas. No caso específico do reconhecimento de sílabas fonéticas, o sistema é treinado para reconhecer e classificar diferentes sílabas com base em suas características acústicas. As outras etapas são classificação das amostras e pós processamento dos dados.

Apesar dos avanços na tecnologia de reconhecimento de fala, o reconhecimento preciso de sílabas fonéticas ainda é um desafio devido à variabilidade natural na fala humana, ruído de fundo e outras condições adversas. No entanto, com o desenvolvimento contínuo de algoritmos de aprendizado de máquina e técnicas de processamento de sinal, espera-se que os sistemas de reconhecimento de sílabas fonéticas continuem melhorando e encontrando novas aplicações em diversas áreas.

Este trabalho validou também o uso da rede CollabNet para o reconhecimento de sílabas fonéticas na língua portuguesa do Brasil. Existem poucas bases de dados fonéticas e poucas ferramentas de processamento de linguagem natural, assim como uma literatura voltadas ao processamento de voz do idioma português, seja ele de Portugal, do Brasil ou de outro país lusófono (SÁ, 2021). Deste modo, o uso do Praat, Kaldi e UFPAlign foram essenciais para a finalização desta pesquisa.

Os resultados obtidos demonstraram que a combinação da técnica de centroides com o uso da rede CollabNet permitiram que sílabas fonéticas fossem classificadas corretamente. Obteve-se uma acurácia média de 75.96% para os dados de áudio da Male/Female. A precisão média ponderada foi de 82%. A sensibilidade média ponderada foi de 76% e o f-score médio ponderado foi de 77%.

Uma desvantagem do método é que ao se utilizar a CollabNet, além dos hiperparâmetros tradicionais (taxa de aprendizagem, quantidade de neurônios em cada camada, entre outros)

que precisam ser definidos e ajustados ao se trabalhar com redes neurais, têm-se mais cinco hiperparâmetros a serem ajustados: Δc, ΔD, $salto_c$, $salto_D$ e a defasagem entre a variação da máscara D e da variável c. Por exemplo, a parametrização correta de ΔD é importante para o treinamento, pois esse parâmetro em conjunto com a variável c administra a atuação da nova camada inserida no treinamento. Uma parametrização não equilibrada desses parâmetros gera o aumento do MSE, por conseguinte, ocasiona prejuízo a aprendizagem. Como exemplo, têm-se os casos em que ΔD é alto, ocasionando uma mudança brusca de valores na máscara D de modo rápido e não permitindo que a rede se adapte por meio da retropropagação do erro a tempo, fazendo com que o MSE suba. Apesar de ter-se esses novos cinco hiperparâmetros, não foi algo que impediu de se ter bons resultados quando se tem uma configuração harmônica.

7.1 TRABALHOS FUTUROS

O conhecimento obtido permitirá realizar pesquisas nessa área, propondo melhorias nos algoritmos, uso de heurísticas e possíveis aplicações reais.

Um dos caminhos a testar é o uso de centroides combinado com outras representações para o reconhecimento de fonemas. Também testar o uso de centroide espectral a partir da transformada Wavelet ou a partir da transformada de Fourier.

O uso de centroides como representação do som pode ser aplicado para o reconhecimento de elementos sonoros de outras naturezas. Isso pode ser alcançado utilizando bases de dados que contenham anotações temporais da ocorrência de efeitos sonoros. Essas anotações permitem o treinamento de algoritmos para reconhecer e classificar esses elementos sonoros com base em suas características acústicas.

Outra área de estudo e aplicação seria no reconhecimento de emoções por meio do som. Trata-se de uma área de pesquisa e aplicação crescente que visa a identificar e entender as emoções expressas na fala ou em outros sinais sonoros. Isso é feito através da análise das características acústicas do som, como tom de voz, ritmo, intensidade e outras qualidades prosódicas.

REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, A.; CARVALHO, F.; MENINO, F. **Introdução ao Machine Learning**. [*S. l.*]: Dataat, 2017. Disponível em: http://dataat.github.io/introducao-ao-machine-learning. Acesso em: 07 out. 2021.

ALMEIDA NETO, Areolino de. **Aplicações de Múltiplas Redes Neurais em Sistemas Mecatrônicos.** 2003. 155 p. Tese (Doutorado em Engenharia Mecânica e Aeronáutica) — Programa de Pós-Graduação em Engenharia Mecânica e Aeronáutica, Divisão de Engenharia Mecânica-Aeronáutica, Instituto Tecnológico de Aeronáutica, São José dos Campos, 2003. Disponível em: http://www.bd.bibl.ita.br/tde_busca/arquivo.php?codArquivo=2732. Acesso em: 15 mar. 2021.

AMBIKAIRAJAH, E. *et al.* Language identification: A tutorial. **IEEE Circuits and Systems Magazine**, [*S. l.*], v. 11, n. 2, p. 82–108, 2011. Disponível em: http://doi.org/10.1109/MCAS.2011.941081. Acesso em: 16 dez. 2021.

AOUANI, H.; BEN, Y. A. Emotion recognition in speech using mfcc with svm, dsvm and auto-encoder. *In*: INTERNATIONAL CONFERENCE ON ADVANCED TECHNOLOGIES FOR SIGNAL AND IMAGE PROCESSING, 4., 2018 Sousse, Tunisia. **Proceedings [...]**. Sousse, Tunisia: IEEE, 2018. p. 1–5. Disponível em: http://doi.org/10.1109/ATSIP.2018.8364518. Acesso em: 09 out. 2021.

AZEVEDO, L. P. de. **Aplicação de redes neurais artificiais no processo de classificação de orquídeas do gênero Cattleya.** 2016. 48p. Monografia (Bacharelado em Sistemas de Informação) — Instituto Federal de Minas Gerais, Sabará, 2016. Disponível em: http://www.ifmg.edu.br/sabara/biblioteca/trabalhos-de-conclusao-de-curso/tcc-documentos/TCCLucasAzevedo.pdf. Acesso em: 10 ago. 2021.

BARRETO, J. M. **Inteligência artificial no limiar do século XXI**. Florianópolis: PPP edições, 1999.

BATISTA, C.; DIAS, A. L.; SAMPAIO NETO, N. C. Free resources for forced phonetic alignment in brazilian portuguese based on kaldi toolkit. **EURASIP Journal on Advances in Signal Processing**, [*S. l.*], v. 2022, p. 1–11, 2022. Disponível em: http://doi.org/10.1186/s13634-022-00844-9. Acesso em: 10 jan. 2023.

BENGIO, Y. *et al.* Greedy layer-wise training of deep networks. *In*: INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 19., 2006, Canada. **Proceedings [...]**. Cambridge, MA, USA: MIT Press, 2006. v. 19, p. 153–160. Disponível em: http://proceedings.neurips.cc/paper_files/paper/2006/file/5da713a690c067105aeb2fae32403405-Paper.pdf. Acesso em: 10 out. 2021.

CAMASTRA, F.; VINCIARELLI, A. **Machine learning for audio, image and video analysis**: theory and applications. 2. ed. Berlin: Springer, 2015.

CAMPOS, F. A. B. **Solução numérica de equações diferenciais parciais via redes neurais artificiais de Legendre e Chebyschev.** 2018. 59p. Dissertação (Mestrado em Matemática e Estatística) — Programa de Pós-Graduação em Matemática e Estatística (PPGME), Universidade

Federal do Pará, Belém, 2018. Disponível em: http://ppgme.propesp.ufpa.br/ARQUIVOS/dissertacoes/2018/Fernando%20Augusto%20Bessa%20Campos.pdf. Acesso em: 15 mai. 2021.

CARDONA, D. A. B.; NEDJAH, N.; MOURELLE, L. M. Online phoneme recognition using multi-layer perceptron networks combined with recurrent non-linear autoregressive neural networks with exogenous inputs. **Neurocomputing**, [*S. l.*], v. 265, p. 78–90, 2017. Disponível em: http://doi.org/10.1016/j.neucom.2016.09.140. Acesso em: 21 dez. 2021.

CARVALHO, M. B. F. de. **Reconhecimento Automático de Fonemas via RNA Profunda.** 2020. 68 p. Dissertação (Mestrado em Ciência da Computação) — Programa de Pós-Graduação em Ciência da Computação/CCET, Universidade Federal do Maranhão, São Luís, 2020. Disponível em: http://tedebc.ufma.br/jspui/handle/tede/tede/3355. Acesso em: 20 abr. 2021.

CATETE, M. R. A.; ROSA, M. de O. Reconhecedor de vogais sustentadas para fala normal e disfônica. **Brazilian Journal of Development**, [*S. l.*], v. 9, n. 05, p. 16394–16400, 05 mai. 2023. Disponível em: http://doi.org/10.34117/bjdv9n5-127. Acesso em: 16 set. 2023.

CHAUDHARY, S. A high-level guide to autoencoders. 21 ago. 2019. Disponível em: http://towardsdatascience.com/a-high-level-guide-to-autoencoders-b103ccd45924. Acesso em: 12 dez. 2021.

CONIAM, D. Voice recognition software accuracy with second language speakers of english. **System**, [*S. l.*], v. 27, n. 1, p. 49–64, 1999. Disponível em: http://doi.org/10.1016/S0346-251X(98)00049-9. Acesso em: 16 dez. 2021.

CUNHA, C.; CINTRA, L. **Nova gramática do português contemporâneo.** 7. ed. Rio de Janeiro: Lexikon, 2016.

DATA SCIENCE ACADEMY. **Deep Learning Book**. [*S. l.*: *s.n.*], 2021. Disponível em: http://www.deeplearningbook.com.br/. Acesso em: 19 out. 2021.

DENG, L.; YU, D. Deep convex net: A scalable architecture for speech pattern classification. *In*: ANNUAL CONFERENCE OF THE INTERNATIONAL SPEECH COMMUNICATION ASSOCIATION, 12., 2011, Florence. **Proceedings [...]**. Florence, Italy: ISCA, 2011. Disponível em: http://www.isca-archive.org/interspeech_2011/deng11_interspeech.pdf. Acesso em: 05 ago. 2021.

DENG, L.; YU, D. **Deep Learning**: Methods and applications. [*S. l.*]: Foundations and trends® in signal processing, 2014. v. 7. Disponível em: http://doi.org/10.1561/2000000039. Acesso em: 09 out. 2021.

DIJKSTRA, B. A. **Reconhecimento de fonemas utilizando redes neurais convolucionais para transcrição fonética automática.** 2021. 76p. Dissertação (Mestrado em Ciência da Computação) — Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2021. Disponível em: http://repositorio.utfpr.edu.br/jspui/bitstream/1/24493/1/reconhecimentofonemasredesneuraisconvolucionais.pdf. Acesso em: 11 out. 2022.

DING, C.; XIE, L.; ZHU, P. Head motion synthesis from speech using deep neural networks. **Multimedia Tools and Applications**, [*S. l.*], v. 74, p. 9871–9888, 2015. Disponível em: http://doi.org/10.1007/s11042-014-2156-2. Acesso em: 16 dez. 2021.

FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, [*S. l.*], v. 27, n. 8, p. 861–874, 2006. Disponível em: http://doi.org/10.1016/j.patrec.2005.10.010. Acesso em: 10 set. 2021.

FLAUZINO, R.A.; SILVA, I.N; SPATTI, D.H. **Redes Neurais Artificiais para Engenharia e Ciências Aplicadas**: Fundamentos teóricos e aspectos práticos. São Paulo: Artliber, 2010.

FURUI, S. Cepstral analysis technique for automatic speaker verification. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, [*S. l.*], v. 29, n. 2, p. 254–272, 1981. Disponível em: http://doi.org/10.1109/TASSP.1981.1163530. Acesso em: 16 dez. 2021.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. *In*: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS, 13., 2010, Sardinia. **Proceedings [...]**. Chia Laguna Resort, Sardinia, Italy: PMLR, 2010. v. 9, p. 249–256. Disponível em: http://proceedings.mlr.press/v9/glorot10a.html. Acesso em: 09 out. 2021.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de imagens digitais**. 3. ed. São Paulo: Blucher, 2009.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. Cambridge, MA, USA: MIT Press, 2016. Disponível em: http://www.deeplearningbook.org. Acesso em: 07 out. 2021.

GREY, J. M.; GORDON, J. W. Perceptual effects of spectral modifications on musical timbres. **The Journal of the Acoustical Society of America**, [*S. l.*], v. 63, n. 5, p. 1493–1500, 1978.

HAYKIN, S.S. **Redes Neurais.** 2. ed. Porto Alegre: Bookman, 2001. Disponível em: http://books.google.com.br/books?id=lBp0X5qfyjUC. Acesso em: 09 out. 2021.

HINTON, G. E.; OSINDERO, S.; TEH, Y. A fast learning algorithm for deep belief nets. **Neural computation**, Cambridge, MA, USA, v. 18, n. 7, p. 1527–1554, 2006.

HUA, Y.; GUO, J.; ZHAO, H. Deep belief networks and deep learning. *In*: INTERNATIONAL CONFERENCE ON INTELLIGENT COMPUTING AND INTERNET OF THINGS, 2015. Harbin. **Proceedings [...]**. Harbin: IEEE, 2015. p. 1–4. Disponível em: http://doi.org/10.1109/ICAIOT.2015.7111524. Acesso em: 10 out. 2021.

KAMARUDIN, N. *et al.* Feature extraction using spectral centroid and mel frequency cepstral coefficient for quranic accent automatic identification. *In*: IEEE STUDENT CONFERENCE ON RESEARCH AND DEVELOPMENT, 2014. **Proceedings [...]**. Penang, Malaysia: IEEE, 2014. p. 1–6. Disponível em: http://doi.org/10.1109/SCORED.2014.7072945. Acesso em: 09 out. 2021.

KHANDELWAL, R. Deep learning — deep boltzmann machine (dbm). 16 dez. 2018. Disponível em: http://medium.datadriveninvestor.com/deep-learning-deep-boltzmann-machine-dbm-e3253bb95d0f. Acesso em: 14 dez. 2021.

LATHI, B. P. **Sinais e sistemas lineares**. 2. ed. Porto Alegre: Bookman, 2006.

LEE, K. F.; HON, H. W. Speaker-independent phone recognition using hidden markov models. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, [*S. l.*], v. 37, n. 11, p. 1641–1648, 1989. Disponível em: http://doi.org/10.1109/29.46546. Acesso em: 16 dez. 2021.

LEE, W. S. *et al.* Fast frequency discrimination and phoneme recognition using a biomimetic membrane coupled to a neural network. **Bioinspiration & Biomimetics**, [*S. l.*], v. 16, n. 2, p. 1–7, 2021. Disponível em: http://doi.org/10.1088/1748-3190/abc869. Acesso em: 10 jan. 2022.

LIMA JUNIOR, M. L. de F. **Deep CollabNet: Rede Deep Learning Colaborativa.** 2018. 51 p. Dissertação (Mestrado em Ciência da Computação) — Programa de Pós-Graduação em Ciência da Computação/CCET, Universidade Federal do Maranhão, São Luís, 2018. Disponível em: http://tedebc.ufma.br/jspui/handle/tede/tede/2318. Acesso em: 25 abr. 2021.

LIU, F.; XU, F.; YANG, S. A flood forecasting model based on deep learning algorithm via integrating stacked autoencoders with bp neural network. *In*: INTERNATIONAL CONFERENCE ON MULTIMEDIA BIG DATA, 3., 2017, Laguna Hills, CA, USA. **Proceedings [...]**. Laguna Hills, CA, USA: IEEE, 2017. p. 58–61. Disponível em: http://doi.org/10.1109/BigMM.2017.29. Acesso em: 10 out. 2021.

LUFT, J. A. **Reconhecimento automático de voz para palavras isoladas e independente do locutor.** 1994. Dissertação (Mestrado em Engenharia Metalúrgica e dos Materiais) — Programa de Pós-Graduação em Engenharia Metalúrgica e dos Materiais, Escola de Engenharia, Universidade Federal do Rio Grande do Sul, 1994. Disponível em: http://hdl.handle.net/10183/189690. Acesso em: 11 out. 2021.

MATUCK, G. R.; SILVA, J. D. S. da. Processamento de sinais de voz-padrões comportamentais por redes neurais artificiais. **Instituto Nacional de Pesquisas Espaciais**, São José dos Campos, p. 1–56, 2005. Disponível em: http://mtc-m21c.sid.inpe.br/attachment.cgi/sid.inpe.br/mtc-m21c/2020/08.03.18.54/doc/processamento%20sinais.pdf. Acesso em: 20 dez. 2021.

MEFTAH, A.; ALOTAIBI, Y. A.; SELOUANI, S. A comparative study of different speech features for arabic phonemes classification. *In*: EUROPEAN MODELLING SYMPOSIUM, 2016. **Proceedings [...]**. [*S. l.*]: IEEE, 2016. p. 47–52.

MEILLET, A. **Linguistique historique et linguistique générale**. Paris: H. Champion, 1926. v. 8.

NASCIMENTO, T. P. **Um serviço baseado em algoritmos genéticos para predição da bolsa de valores.** 2015. 61 p. Dissertação (Mestrado em Engenharia de Eletricidade) — Programa de Pós-Graduação em Engenharia de Eletricidade/CCET, Universidade Federal do Maranhão, São Luís, 2015. Disponível em: http://tedebc.ufma.br/jspui/handle/tede/290. Acesso em: 27 nov. 2021.

NEDJAH, N.; BONILLA, A. D.; MOURELLE, L. de M. Automatic speech recognition of portuguese phonemes using neural networks ensemble. **Expert Systems with Applications**, [*S. l.*], v. 229, p. 1–23, 2023. Disponível em: http://doi.org/10.1016/j.eswa.2023.120378. Acesso em: 16 out. 2023.

NIQUINI, F. M. M. **Reconhecimento de comandos de voz com verificação de locutores e vocabulário restrito utilizando redes neurais artificiais.** 2007. 50p. Monografia (Bacharelado em Engenharia Elétrica) — Universidade Federal de Viçosa, Viçosa, 2007. Disponível em: http://www3.dti.ufv.br/sig_del/consultar/monografia. Acesso em: 24 mar. 2023.

OLIVEIRA NETO, J. A. de; CASTRO, M. A. A.; FELIX, L. B. Reconhecimento de comandos de voz para o acionamento de cadeira de rodas. *In*: CONGRESSO BRASILEIRO DE AUTOMÁTICA, 15., 2010. Bonito. **Anais [...]**. Bonito, MS: [s.n.], 2010. p. 3819–3824.

PARCOLLET, T. *et al.* Quaternion convolutional neural networks for end-to-end automatic speech recognition. **arXiv**, [*S. l.*], p. 1–5, 2018. Disponível em: http://arxiv.org/abs/1806.07789. Acesso em: 16 dez. 2021.

PEETERS, G. A large set of audio features for sound description (similarity and classification) in the cuidado project. **IRCAM**, Paris, France, v. 54, n. 1, p. 1–25, 2004. Disponível em: http://recherche.ircam.fr/equipes/analyse-synthese/peeters/ARTICLES/Peeters_2003_cuidadoaudiofeatures.pdf. Acesso em: 16 dez. 2021.

PETRY, A.; ZANUZ, A.; BARONE, D. A. C. Utilização de técnicas de processamento digital de sinais para a identificação automática de pessoas pela voz. *In*: SIMPÓSIO SOBRE SEGURANÇA EM INFORMÁTICA, 1999. São José dos Campos. **Palestras [...]**. São José dos Campos, SP: Academia edu, 1999. p. 1–7. Disponível em: http://www.academia.edu/download/35397469/utilizacao_de_tecnicas_para_reconhecimento_automatico_do_falante.pdf. Acesso em: 10 out. 2021.

POVEY, D. *et al.* The kaldi speech recognition toolkit. *In*: IEEE 2011 WORKSHOP ON AUTOMATIC SPEECH RECOGNITION AND UNDERSTANDING, 2011, Big Island. **Proceedings [...]**. Big Island, Hawaii, US: IEEE Signal Processing Society, 2011.

PULAVARTI, S. V. S. R. K. *et al.* From protein design to the energy landscape of a cold unfolding protein. **The Journal of Physical Chemistry B**, [*S. l.*], v. 126, n. 6, p. 1212–1231, 2022.

QUINTANILHA, I. M.; BISCAINHO, L. W. P.; LIMA NETTO, S. Towards an end-to-end speech recognizer for portuguese using deep neural networks. *In*: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS, 35., 2017, São Pedro. **Anais [...]**. São Pedro, SP: SBrT, 2017. p. 709–714.

RABINER, L. R. A tutorial on hidden markov models and selected applications in speech recognition. **Proceedings of the IEEE**, [*S. l.*], v. 77, n. 2, p. 257–286, 1989. Disponível em: http://doi.org/10.1109/5.18626. Acesso em: 16 dez. 2021.

RABINER, L. R.; SAMBUR, M. R. An algorithm for determining the endpoints of isolated utterances. **Bell System Technical Journal**, [*S. l.*], v. 54, n. 2, p. 297–315, 1975. Disponível em: http://doi.org/10.1002/j.1538-7305.1975.tb02840.x. Acesso em: 15 dez. 2021.

ROQUE, T. R.; MENDES, R. S. Extraçao de centroide espectral através da tranformada wavelet packet. *In*: SPS, 2013. **Proceedings [...]**. [*S. l.*]: [s.n.], 2013. v. 2013.

SÁ, J. M. A. M. de. **Reconhecimento de fala em português de Portugal num contexto com poucos recursos.** 2021. 95 p. Dissertação (Mestrado Integrado em Engenharia de Redes e Sistemas Informáticos) — Departamento de Ciências de Computadores, Faculdade de Ciências da Universidade do Porto, Porto, 2021. Disponível em: http://hdl.handle.net/10216/139258. Acesso em: 25 abr. 2022.

SALAKHUTDINOV, R.; HINTON, G. Deep boltzmann machines. *In*: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND STATISTICS, 12., 2009, Florida. **Proceedings [...]**. Hilton Clearwater Beach Resort, Clearwater Beach, Florida, USA: PMLR, 2009. v. 5, p. 448–455. Disponível em: http://proceedings.mlr.press/v5/salakhutdinov09a.html. Acesso em: 03 ago. 2021.

SAMPAIO NETO, N. C. **Ferramentas e recursos livres para reconhecimento e síntese de voz em português brasileiro.** 2011. 96p. Tese (Doutorado em Engenharia Elétrica) — Universidade Federal do Pará, Belém, 2011. Disponível em: http://repositorio.ufpa.br/jspui/handle/2011/2845. Acesso em: 10 out. 2021.

SAMPAIO NETO, N. C. *et al.* Spoltech and ogi-22 baseline systems for speech recognition in brazilian portuguese. *In*: COMPUTATIONAL PROCESSING OF THE PORTUGUESE LANGUAGE: INTERNATIONAL CONFERENCE, 8., 2008, Aveiro, Portugal. **Proceedings [...]**. Berlin, Heidelberg: Springer, 2008. p. 256–259. Disponível em: http://doi.org/10.1007/978-3-540-85980-2_33. Acesso em: 09 out. 2021.

SHARMA, A. Restricted boltzmann machines — simplified. 01 out. 2018. Disponível em: http://towardsdatascience.com/restricted-boltzmann-machines-simplified-eab1e5878976. Acesso em: 20 dez. 2021.

SILVA, M. T.; SANTOS, C. M. D. Uma análise histórica sobre a seleção natural: de darwin-wallace à síntese estendida da evolução. **Amazônia: Revista de Educação em Ciências e Matemáticas**, Belém, PA, v. 11, n. 22, p. 46–61, 2015.

SLAVIERO, D.E. **Uso de características significativas em sistema de identificação de língua em música.** 2006. 62 p. Monografia (Bacharelado em Ciência da Computação) — Universidade de Caxias do Sul, Caxias do Sul, 2022. Disponível em: http://repositorio.ucs.br/11338/11991. Acesso em: 10 jan. 2023.

SOLTI, E. **Relações entre o ensino de idiomas e estudo do jazz: uma proposta de aplicativo digital para desenvolvimento da expressividade musical na guitarra.** 2022. 154 p. Tese (Doutorado em Música) — Universidade de Estadual de Campinas, Campinas, 2022. Disponível em: http://repositorio.unicamp.br/Busca/Download?codigoArquivo=554734. Acesso em: 08 jan. 2023.

VALIATI, J. F. **Reconhecimento de voz para comandos de direcionamento por meio de redes neurais**. Dissertação (Mestrado em Computação) — Programa de Pós-Graduação em Computação. Instituto de Informática. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2000. Disponível em: http://hdl.handle.net/10183/2947. Acesso em: 11 set. 2021.

VIANA, F. dos S. **Redes neurais aplicadas na estratégia de ponderação de partículas em SLAM.** 2019. 75p. Dissertação (Mestrado em Engenharia de Computação e Sistemas) — Programa de Pós-Graduação em Engenharia de Computação e Sistemas, Universidade Estadual do Maranhão, São Luís, 2019. Disponível em: http://repositorio.uema.br/handle/123456789/1253. Acesso em: 08 dez. 2021.

VINCENT, P. *et al.* Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. **Journal of machine learning research**, [*S. l.*], v. 11, n. 12, p. 3371–3408, 2010. Disponível em: http://jmlr.org/papers/v11/vincent10a.html. Acesso em: 16 dez. 2021.

WEISS, C. S. *et al.* **Comunicação e linguagem**. Indaial: Uniasselvi, 2018.

ZHANG, Y. *et al.* Towards end-to-end speech recognition with deep convolutional neural networks. **arXiv**, [*S. l.*], p. 1–5, 2017. Disponível em: http://arxiv.org/abs/1701.02720. Acesso em: 16 dez. 2021.

APÊNDICE A – SINAL DE FALA

O sinal de fala é resultado de diversos níveis de informação, os quais podem ser utilizados para discriminar línguas (AMBIKAIRAJAH *et al.*, 2011 apud SLAVIERO, 2022):

- Acústica: o nível mais próximo da física original da fala, onde diferentes fenômenos da fala podem ser distinguidos com base na amplitude e frequência das ondas. Informações como fonotática e da palavra podem ser extraídas a partir deste nível, sendo elas um dos tipos de conhecimento mais fáceis de obter do áudio bruto (SLAVIERO, 2022).

- Fonotática: Cada língua tem um número limitado de sons possíveis e a fonotática é o estudo dos padrões sonoros pertinentes de uma língua singular. É um ramo da fonologia, o estudo do sistema de som de uma língua. Existem diferenças nas restrições fonotáticas entre os idiomas, por exemplo, o fonema /st/ é demasiadamente comum no inglês, enquanto no japonês não é aceito (SLAVIERO, 2022).

- Prosódico: Os idiomas variam em ritmo, entonação e intensidade. Durante a fala, assim como as diferenças de duração, tom e volume, todas essas características definem os principais aspectos do que é prosódico. A parte principal da compreensão auditiva humana, esse nível inclui recursos que ajudam a distinguir os idiomas sem entrar no significado das palavras. Por exemplo, idiomas tonais como o chinês mandarim e o vietnamita, onde a inflexão de uma palavra determina seu significado (AMBIKAIRAJAH *et al.*, 2011 apud SLAVIERO, 2022).

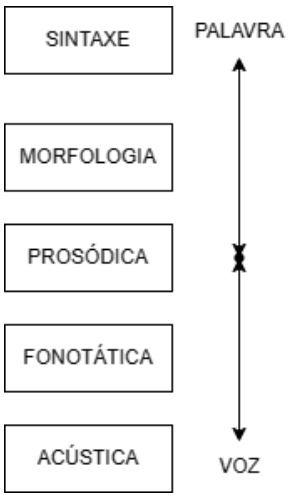
- Morfologia: Cada língua geralmente possui seu próprio conjunto de vocabulário e uma maneira única de formar palavras. A formação de palavras ocorre através de diferentes combinações de fonemas, e a morfologia é a área da linguística que explora a estrutura interna dessas palavras (SLAVIERO, 2022).

- Sintaxe: Quando as pessoas se comunicam, é necessário combinar as palavras de uma maneira específica para criar uma mensagem compreensível. A sintaxe é o estudo dos princípios e regras que determinam como as palavras em uma frase devem ser organizadas para transmitir um significado. Todas as línguas possuem uma variedade de regras sintáticas para organizar a estrutura das palavras, formando assim frases e sentenças (SLAVIERO, 2022).

Esses níveis podem ser ainda agrupados, como apresentado na Figura 44, em dois níveis mais amplos: um nível mais próximo da voz, no qual as características podem ser obtidas a partir de dados de áudio brutos, e o nível da palavra, no qual é possível diferenciar as línguas com base em suas palavras e vocabulários distintos. O nível da voz abrange características acústicas,

fonéticas, fonotáticas e prosódicas. Por outro lado, o nível da palavra aborda características como vocabulário, sintaxe, morfologia e gramática (SLAVIERO, 2022).

Figura 44 – Diferentes níveis de características na fala e na língua.



Fonte: Adaptado de Ambikairajah *et al.* (2011)

APÊNDICE B – SISTEMA DE FALA

A voz está associada ao som produzido pela vibração das pregas vocais, enquanto a fala envolve a articulação dos sons na boca por meio dos articuladores. A fala está relacionada às palavras, e a voz e a fala estão interligadas.

Os sons da nossa fala são, em sua maioria, resultantes da ação de certos órgãos sobre a corrente de ar proveniente dos pulmões. Três condições são necessárias para a sua produção:

- a corrente de ar;
- um obstáculo encontrado por essa corrente de ar;
- uma caixa de ressonância.

Estas condições são criadas pelos órgãos da fala, em conjunto denominados aparelho fonador.

B.1 APARELHO FONADOR

O aparelho fonador, apresentado na Figura 45, é responsável por controlar a passagem do ar, a configuração dos obstáculos e a forma como a caixa de ressonância é utilizada, permitindo assim a produção dos diferentes sons da fala.

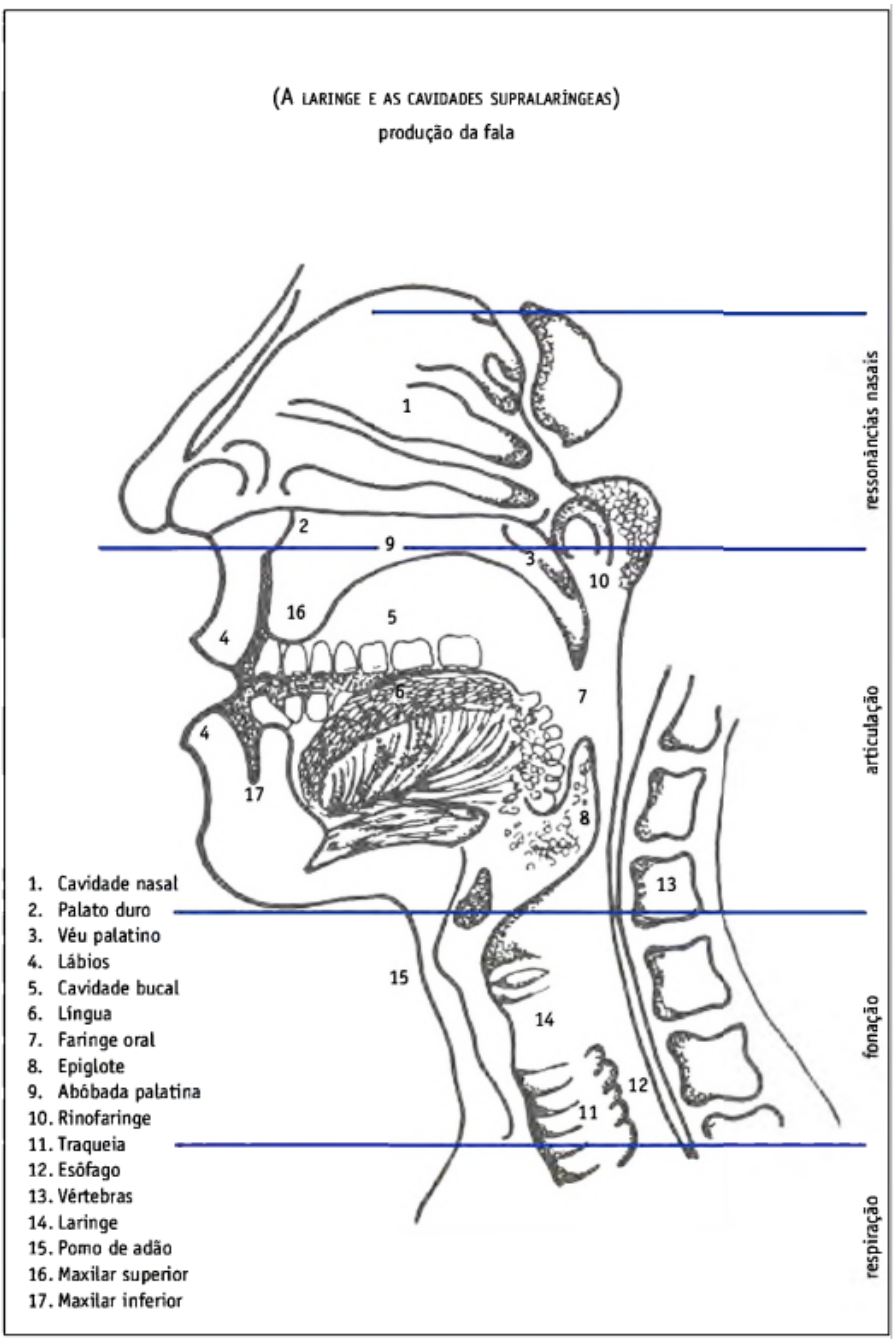
É constituído das seguintes partes:

- os pulmões, os brônquios e a traqueia - órgãos respiratórios que fornecem a corrente de ar, matéria-prima da fonação;
- a laringe, onde se localizam as cordas vocais, que produzem a energia sonora utilizada na fala;
- as cavidades supralaríngeas (faringe, boca e fossas nasais), que funcionam como caixas de ressonância, sendo que a cavidade bucal pode variar profundamente de forma e de volume, graças aos movimentos dos órgãos ativos, sobretudo da língua, que, de tão importante na fonação, se tornou sinônimo de “idioma”.

O ar que é expelido dos pulmões, através dos brônquios, entra na traqueia e alcança a laringe. Na laringe, ao atravessar a glote, geralmente encontra o primeiro obstáculo em seu caminho. A glote está localizada na altura do chamado “pomo de adão” ou “gogó” e consiste na abertura entre duas pregas musculares das paredes superiores da laringe, conhecidas como cordas vocais.

O fluxo de ar pode encontrar a glote fechada ou aberta, dependendo da proximidade ou afastamento dos bordos das cordas vocais. Quando a glote está fechada, o ar é forçado a passar

Figura 45 – O aparelho fonador.



Fonte: Cunha e Cintra (2016)

através das cordas vocais tensas, causando vibrações que resultam no som musical característico das articulações sonoras. Por outro lado, quando as cordas vocais estão relaxadas e a glote está aberta, o ar escapa sem causar vibrações laríngeas, produzindo assim sons chamados de “surdos”.

Em resumo, é nas cordas vocais e na configuração da glote que encontramos a origem dos diferentes tipos de sons produzidos durante a fala e a emissão de sons vocais. A distinção entre os sons sonoros e surdos pode ser claramente percebida na pronúncia de algumas consoantes que, em sua maioria, têm características idênticas, exceto pela vibração das cordas vocais.

Um exemplo comum é a diferença entre as consoantes “b” e “p”, apresentado na Tabela 7.

Tabela 7 – Exemplos de consoantes surdas e sonoras.

Sonora	Surda
/b/	/p/
/d/	/t/
/g/	/k/
/v/	/f/
/z/	/s/

Fonte: Elaborado pela autora (2023).

Ambas são produzidas da mesma forma na boca, envolvendo o fechamento temporário dos lábios e a liberação explosiva do ar. A única diferença é que o som da letra “b” é sonoro, enquanto o som da letra “p” é surdo.

Quando pronunciamos o “b”, as cordas vocais vibram durante a emissão do som, o que cria a característica sonora. Já no caso do “p”, não há vibração das cordas vocais, tornando-o um som surdo.

O mesmo se aplica a outras consoantes parecidas, como “d” e “t”, “g” e “k”, “v” e “f”, entre outras, apresentadas na Tabela 7. A presença ou ausência da vibração das cordas vocais é o que diferencia esses sons e os torna sonoros ou surdos. Essa distinção é fundamental na formação correta das palavras em muitos idiomas, incluindo a língua portuguesa.

Quando a corrente de ar expiratória sai da laringe, ela entra na cavidade faríngea, uma encruzilhada que oferece duas vias de acesso ao exterior: o canal bucal e o nasal. No ponto onde esses dois canais se cruzam, encontra-se o véu palatino, um órgão com mobilidade capaz de obstruir ou permitir o ingresso do ar na cavidade nasal, determinando, assim, se um som é oral ou nasal.

Quando o véu palatino é levantado, ele adere à parede posterior da faringe, permitindo apenas o fluxo de ar pelo canal bucal. As articulações produzidas dessa maneira são chamadas de "orais". Por outro lado, quando o véu palatino é abaixado, ambos os canais ficam livres. A corrente de ar é então dividida e parte dela passa pelas fossas nasais, onde adquire a ressonância característica das articulações, sendo, por esse motivo, chamadas de "nasais".

Todavia, é na cavidade bucal que são produzidos os movimentos fonadores mais variados, graças à maior ou menor separação dos maxilares, das bochechas e, principalmente, à mobilidade da língua e dos lábios.

Tabela 8 – Exemplos de sons orais e nasais.

Oral	Nasal
/a/	/ã/
/k/	/m/

Fonte: Elaborado pela autora (2023).

Ao se comparar a pronúncia das vogais /a/ e /ã/, apresentadas como exemplo na Tabela 8, em palavras como: lá e lã, mato e manto. Sabe-se que as palavras “lá” e “mato” são classificadas

como orais, pois /a/ é um som oral. Já as palavras “lã” e “manto” são nasais, pois /ã/ é um som nasal.

B.2 DESCRIÇÃO FONÉTICA E FONOLÓGICA.

A descrição dos sons da fala (descrição fonética), para ser completa, deveria considerar sempre:

a) como eles são produzidos;

b) como são transmitidos;

c) como são percebidos.

O interesse principal da descrição deveria concentrar-se na impressão auditiva, pois é através dela que podemos perceber a variedade dos sons e entender como eles funcionam na representação dos fonemas. A descrição fonológica é amplamente fundamentada na base acústica, e é por meio da percepção auditiva que podemos apreciar e compreender as nuances dos sons da fala.

Através da impressão auditiva, somos capazes de distinguir os diferentes fonemas presentes em um idioma, percebendo as variações sutis de sons que podem alterar o significado das palavras. É a partir dessa percepção que podemos analisar e descrever a organização dos sons em um sistema fonológico específico de uma língua.

Dessa forma, a percepção auditiva desempenha um papel fundamental na descrição fonológica, pois é através dela que podemos compreender e interpretar como os sons são produzidos e organizados na fala, o que nos permite estudar e apreciar a riqueza e complexidade dos sistemas fonológicos das diversas línguas.

De fato, a descrição do efeito acústico de um fonema não é feita com termos precisos, semelhantes àqueles usados para descrever os movimentos dos órgãos envolvidos na produção de um som. Os avanços na fonética acústica são, aliás, relativamente recentes (CUNHA; CINTRA, 2016).

Enquanto a articulação dos sons da fala pode ser descrita de forma mais detalhada, descrever os sons com base em suas propriedades acústicas é um campo de estudo que se desenvolveu mais recentemente. A fonética acústica envolve a análise das características físicas dos sons da fala, como frequência, intensidade e duração, que são captadas por meio de equipamentos especializados, como microfones e espectrografias.

Essa abordagem acústica permite uma análise mais objetiva e quantitativa dos sons da fala, proporcionando uma compreensão mais profunda das variações sonoras e das distinções entre os fonemas em diferentes idiomas. Com o avanço da tecnologia, a fonética acústica tem contribuído significativamente para a pesquisa em linguística e para a compreensão da produção e percepção dos sons da fala.

A fonética fisiológica, que se baseia na articulação dos sons da fala, é uma especialidade antiga e amplamente desenvolvida, pois os órgãos fonadores e seu funcionamento são bem compreendidos. Por isso, os fonemas são frequentemente descritos e classificados com base em suas características articulatórias. No entanto, observa-se uma tendência moderna de associar a descrição acústica à fisiológica ou de realizá-las em paralelo.

Essa abordagem fisiológica permite uma análise detalhada da produção dos sons da fala, descrevendo como os órgãos articulatórios, como a língua, os lábios e os dentes, se movem para produzir os diferentes sons. Essa compreensão da articulação é fundamental para a identificação e classificação dos fonemas em diferentes idiomas.

No entanto, à medida que a tecnologia avança, a descrição acústica dos sons da fala também ganha destaque. A análise das características acústicas, como frequência, intensidade e duração dos sons, complementa a descrição fisiológica, permitindo uma compreensão mais completa dos fonemas e suas variações sonoras.

Assim, a fonética fisiológica e a fonética acústica podem ser vistas como abordagens complementares, que juntas fornecem uma visão mais abrangente e aprofundada da produção e percepção dos sons da fala em estudos linguísticos.

APÊNDICE C – INTELIGÊNCIA ARTIFICIAL

A inteligência artificial (IA) pode ser considerada como uma tentativa de transferir o aprendizado e o pensamento humano para o computador, dando-lhe inteligência. Em vez de ser programada para todos os fins, uma IA pode encontrar respostas e resolver problemas de forma independente (NASCIMENTO, 2015).

No caso da inteligência artificial, o programador não especifica o que a aplicação faz em cada etapa individual, mas escreve um algoritmo capaz de aprender de forma independente e, assim, lidar com dados previamente desconhecidos, encontrar padrões ou agir. As IAs são, portanto, muito mais poderosas que os sistemas baseados em regras, porque podem - até certo ponto - reagir a situações anteriormente desconhecidas e aprender com a experiência.

A IA se compromete com algumas abordagens que possuem seu devido valor, são elas: abordagem simbólica, abordagem evolutiva, abordagem conexionista. Na abordagem simbólica, é investigada a cognição, tendo como objetivo desenvolver o aprendizado a partir de logica de predicados e regras de produção, por exemplo, os sistemas especialistas. Assim, na abordagem evolutiva, é baseada nos princípios da seleção natural e definições genéticas, desenvolvendo modelos computacionais com objetivo de otimização. Os principais exemplos desse paradigma são os algoritmos genéticos e outras metas heurísticas. Por fim, na abordagem conexionista, é investigado o funcionamento do cérebro humano para as suas modelagens computacionais, acreditando que as condutas dos neurônios é a inteligência e, que eles geram o aprendizado, um exemplo, são as redes neurais artificiais (BARRETO, 1999; NASCIMENTO, 2015).

C.1 *MACHINE LEARNING*

O aprendizado de máquina (na língua inglesa, *machine learning* - ML) é um âmbito da IA que utiliza algoritmos para extrair informações de dados brutos e representá-los através de algum padrão matemático, propiciando ao sistema computacional a capacidade de aprender, identificar padrões e exercer a tomada de decisão sem a interferência do usuário. Este padrão é utilizado para fazer inferências a partir de outros conjuntos de dados. Existem muitos algoritmos que permitem encontrar este padrão, mas um tipo em especial que tem destaque são as redes neurais artificiais (GOODFELLOW *et al.*, 2016; ALMEIDA *et al.*, 2017).

O campo do aprendizado de máquina abrange uma ampla gama de tópicos e possui várias subáreas de aplicação. No entanto, para fins de organização, os métodos de aprendizado são agrupados em categorias principais, que dependem dos dados utilizados e da estratégia empregada pelos algoritmos. As principais categorias de aprendizado de máquina são: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço, conforme ilustrado na Figura 46.

- **Aprendizado supervisionado**: Esta abordagem utiliza dados “rotulados” ou “etique-

Figura 46 – Diferentes categorias de aprendizagem.

Fonte: Almeida *et al.* (2017)

tados", fornecidos por um "professor". Os dados têm rótulos previamente atribuídos, podendo ser de maneira manual ou automática. O objetivo é desenvolver um modelo capaz de relacionar os dados de entradas com as etiquetas de saídas através do treinamento com um conjunto de dados de treino. Esse modelo é, então, empregado para rotular novos dados. Exemplos de algoritmos de aprendizado supervisionado incluem SVM, Regressão Linear, Regressão Logística, Árvores de Decisão, Florestas Aleatórias, *k* Vizinhos Mais Próximos, RNAs e Naive Bayes (GOODFELLOW *et al.*, 2016; ALMEIDA *et al.*, 2017).

- **Aprendizado não supervisionado**: Nesta abordagem, não se utiliza dados rotulados e o objetivo é descobrir padrões ou características latentes nos dados que permitam agrupá-los em categorias não definidas previamente. Exemplos de algoritmos nessa abordagem incluem *k*-médias, RNAs, HMM, Análise de Componentes Principais, Análise de Componentes Independentes e Mapas Auto-Organizáveis (GOODFELLOW *et al.*, 2016; ALMEIDA *et al.*, 2017).

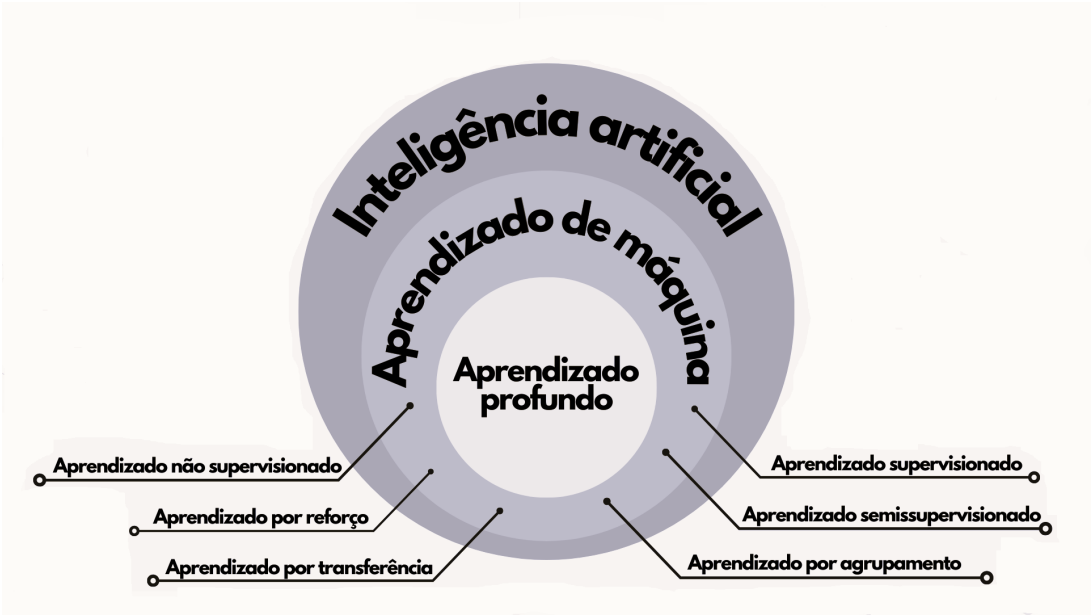
- **Aprendizado por reforço**: Nesta abordagem, é fundamentado na interação entre um agente e um ambiente. O agente toma decisões com base em uma política dentro desse ambiente, o qual influencia as transições de estados e as recompensas obtidas a cada ação executada. O objetivo é encontrar uma política ótima que maximize as recompensas

obtidas pelo agente (GOODFELLOW *et al.*, 2016; ALMEIDA *et al.*, 2017).

As redes neurais artificiais não são absolutamente novas, existem pelo menos desde a década de 1950. Todavia no decurso de algumas décadas, não obstante a arquitetura desses modelos tivesse evoluído, ainda faltavam constitutivos que fizessem os modelos realmente funcionarem. No meio da década de 1980 e no início da década de 1990, muitos avanços importantes na arquitetura das redes neurais artificias ocorreram. Contudo, a quantidade de tempo e de dados necessários para obter bons resultados protelou a adoção e, portanto, o interesse foi arrefecido, com o que ficou conhecido como *AI Winter* (na língua portuguesa, Inverno da IA) (DATA SCIENCE ACADEMY, 2021).

No começo dos anos 2000, o poder computacional cresceu de modo exponencial e o mercado constatou um aumento súbito de técnicas computacionais que não eram possíveis outrora disso. Foi quando o aprendizado profundo (na língua inglesa, *deep learning* - DL) surgiu do crescimento computacional explosivo dessa década como o principal mecanismo de construção de sistemas de IA, ganhando muitas competições importantes de ML. Nos dias atuais o interesse por DL não parou de crescer, o termo "aprendizado profundo" é mencionado com frequência cada vez maior e várias soluções comerciais surgem a todo momento (DATA SCIENCE ACADEMY, 2021).

Figura 47 – IA, ML, DL e tipos de aprendizagem.



Fonte: Elaborado pela autora (2023).

A aprendizagem profunda ou *deep learning* é um subcampo dentro da ML, conforme Figura 47, que é baseado em algoritmos que exploram muitas camadas de processamento de informações não lineares para extração e transformação supervisionada ou não supervisionada de características, e para análise e classificação de padrões.

O DL tem como objetivo modelar relações complexas entre os dados. A essência do aprendizado profundo é computar as características ou representações hierárquicas dos dados observacionais, onde as características ou fatores de nível superior são definidos a partir dos de nível inferior. Os níveis nestes modelos estatísticos aprendidos correspondem a níveis distintos de conceitos, onde os conceitos do nível superior são definidos a partir dos conceitos do nível inferior, e os mesmos conceitos do nível inferior podem ajudar a definir muitos conceitos do nível superior (DENG; YU, 2014).

A família de métodos de aprendizagem profunda tem se tornado cada vez mais rica, abrangendo as redes neurais, modelos probabilísticos hierárquicos e uma variedade de algoritmos de aprendizagem de características não supervisionados e supervisionados. Pela tradição do ML comumente adotada, pode ser natural apenas classificar as técnicas de aprendizagem profunda em modelos profundamente discriminatórios e modelos generativos/não supervisionados.

Modelos profundamente discriminatórios se tem por exemplo, redes neurais profundas (DNNs), redes neurais recorrentes (RNNs), redes neurais convolucionais (CNNs), entre outros. Modelos generativos/não supervisionados se tem por exemplo, *restricted Boltzmann machine* (RBMs), *deep belief networks* (DBNs), *deep Boltzmann machines* (DBMs), *autoencoders* regularizados , e muitos outros (DENG; YU, 2014).