

# Classificação de Doenças Neurodegenerativas Utilizando Dados de EEG

**Disciplina:** MO444 - Aprendizado de Máquina

**Universidade:** Universidade Estadual de Campinas - Instituto de Computação

**Integrantes:**

Betania E R da Silva (167508)

Decio Miranda Filho (236087)

Leticia Balbi Silva (291072)

Rafael Simionato (066505)

**Professora:** Sandra Avila

## Resumo

O trabalho aborda a classificação de doenças neurodegenerativas utilizando sinais de EEG, explorando duas abordagens principais: aprendizado auto-supervisionado e supervisionado. Para isso, foram analisados sinais de EEG de 88 indivíduos, divididos entre pacientes saudáveis (CN), com Alzheimer (AD) e Demência Frontotemporal (FTD). Inicialmente, a abordagem auto-supervisionada foi escolhida para modelar o problema. O algoritmo Contrastive Predictive Coding (CPC) foi aplicado com o objetivo de extrair características relevantes das séries temporais de EEG. Duas cabeças de predição baseadas em MLP foram acopladas ao *backbone* pré-treinado. Já na abordagem supervisionada, foi utilizada uma arquitetura híbrida CNN-LSTM, combinando capacidades de extração de características espaciais e temporais.

Os resultados finais indicaram que o modelo utilizado como *benchmark* ainda apresentou melhor desempenho geral em comparação com as abordagens propostas. Na abordagem com o CPC, o método não foi capaz de aprender representações sobre as classes. Já a combinação CNN-LSTM enfrentou dificuldades com sobreajuste. Apesar de alguns desafios relacionados à generalização, o trabalho reforçou o potencial de ambas as abordagens para análises futuras em sinais de EEG, apontando para melhorias em arquitetura e processamento de dados.

# 1 Introdução

Com os avanços da medicina, especialmente no âmbito da Medicina 2.0, observa-se um aumento na expectativa de vida, conforme argumentado por Peter Attia em "Outlive" [1]. No entanto, a longevidade está frequentemente associada ao surgimento de doenças crônicas, denominadas por Attia como os "quatro cavaleiros": doenças cardiovasculares, câncer, disfunções metabólicas e doenças neurodegenerativas. A lógica que emerge é a necessidade de combate precoce a esses "cavaleiros", o que implica a realização de exames especializados para um diagnóstico antecipado. Entretanto, em países de grande extensão territorial, como o Brasil, o sistema de saúde pública enfrenta desafios de infraestrutura e capacidade. Um exemplo disso é Curitiba, no estado do Paraná, onde o tempo de espera por exames varia entre 7 e 14 meses [2]. Assim, há uma necessidade urgente de explorar métodos tecnológicos acessíveis e de baixo custo que possam ser amplamente implementados no país, facilitando o apoio ao diagnóstico médico.

Nesse contexto, alinhado ao Objetivo de Desenvolvimento Sustentável (ODS) 3 da ONU, que visa assegurar saúde de qualidade e promover o bem-estar para todos, este trabalho propõe a classificação de doenças neurodegenerativas por meio da análise de séries temporais extraídas de sinais obtidos em exames de eletroencefalograma (EEG). O EEG é um exame não invasivo que registra a atividade elétrica do cérebro através de eletrodos posicionados no couro cabeludo do paciente. A coleta dos impulsos elétricos gera padrões que refletem o funcionamento cerebral, permitindo a identificação de condições patológicas. Entre os fatores que influenciam os sinais de EEG estão eventos naturais, como as fases do sono, e condições patológicas, como epilepsia, convulsões, doença de Alzheimer e demência [3]. Neste estudo, o foco será direcionado às doenças de Alzheimer (AD) e à demência frontotemporal (FTD).

O uso de métodos de apoio ao diagnóstico de doenças neurodegenerativas baseados na análise de sinais de EEG apresenta-se como uma alternativa promissora. Esses métodos atendem aos critérios de custo reduzido e ampla disponibilidade, destacando-se por serem mais rápidos em comparação a exames de neuroimagem [4]. Ademais, considerando que não existe cura para as doenças neurodegenerativas, o diagnóstico precoce é fundamental para a implementação de cuidados preventivos, o que pode retardar o agravamento dos sintomas e melhorar a qualidade de vida dos pacientes, especialmente dos idosos, grupo mais afetado por essas condições [5].

Para o desenvolvimento deste trabalho, utilizou-se um conjunto de dados [6] contendo sinais de EEG obtidos em exames de rotina. O objetivo é realizar a classificação binária entre as classes Alzheimer (AD) vs. Saudável (CN) e Demência Frontotemporal (FTD) vs. Saudável (CN). Inicialmente, foi empregada a técnica de aprendizado auto-supervisionado denominada *Contrastive Predictive Coding* (CPC) para a extração de informações relevantes das séries temporais. Posteriormente, avaliou-se a eficácia dessa técnica na classificação das doenças em estudo. Complementarmente, foram aplicadas técnicas de aprendizado supervisionado baseadas no estado da arte para análise de sinais de EEG. Os resultados obtidos são apresentados e discutidos, destacando-se as limitações encontradas e as perspectivas para futuros desenvolvimentos.

## 2 Trabalhos Relacionados

Esta seção discute a literatura relevante para o presente projeto, destacando trabalhos que realizaram abordagens semelhantes.

## 2.1 Abordagens auto-supervisionadas

As técnicas de aprendizado auto-supervisionado (*Self-Supervised Learning* — SSL) visam treinar um extrator de características robusto (*backbone*) por meio de tarefas de pretexto utilizando dados não rotulados [7] [8]. O processo de SSL é geralmente dividido em duas fases: pré-treinamento e ajuste fino (*fine-tuning*). Na fase de pré-treinamento, o modelo aprende padrões relevantes ao resolver uma tarefa pretexto, como a previsão de rotações de imagens ou a reconstrução de partes ausentes. O modelo resultante, composto por um *backbone* e uma cabeça de projeção, é posteriormente refinado na fase de ajuste fino, onde a cabeça de projeção é substituída por uma cabeça de predição alinhada à tarefa-alvo, como classificação ou regressão, utilizando dados rotulados.

Entre as diversas técnicas propostas nesse contexto, a *Contrastive Predictive Coding* (CPC) destaca-se como uma abordagem eficaz. Desenvolvida inicialmente por Oord et al. [9] para imagens, a CPC foi posteriormente aplicada ao reconhecimento de atividades humanas (*Human Activity Recognition*, HAR) [10]. Essa aplicação utiliza séries temporais obtidas por sensores inerciais, como acelerômetros e giroscópios de *smartphones*, demonstrando seu potencial no domínio de séries temporais. Essa técnica será detalhada na Subseção 3.1.

Xiang Zhang et al. [11] também exploraram o SSL, ampliando sua aplicação para diferentes conjuntos de dados. O trabalho avaliou oito *datasets* de séries temporais, incluindo SLEEPEG e EPILEPSY, com foco em tarefas como classificação de estágios de sono, diagnóstico de epilepsia, reconhecimento de atividades, gestos e detecção de falhas mecânicas. Durante o pré-treinamento, o *dataset* SLEEPEG, caracterizado por sua alta complexidade temporal, foi utilizado para treinar o modelo baseado na consistência tempo-frequência (*Time-Frequency Consistency*, TFC). Comparado a dez técnicas do estado da arte, como CPC, SimCLR e TS2vec, o TFC demonstrou superioridade em diversos cenários, destacando-se como uma abordagem promissora para pré-treinamento em dados médicos e outras séries temporais complexas.

## 2.2 Abordagens supervisionadas

Para a abordagem supervisionada, foi realizada uma revisão de literatura com o objetivo de validar a proposta de utilização de extitConvolutional Neural Networks (CNN) combinadas com extitLong Short-Term Memory (LSTM). Zhao et al. (2019) destacam a eficácia dessa arquitetura híbrida para o reconhecimento de emoções a partir de falas, aproveitando a capacidade das CNNs de extrair representações robustas de características e o potencial das LSTMs para capturar padrões temporais [12].

De forma semelhante, Mutegeki et al. (2020) investigaram a aplicação de redes CNN-LSTM no reconhecimento de atividades humanas a partir de dados de sensores, demonstrando a robustez do modelo [13]. Outros estudos que ressaltam a importância da combinação de características temporais e espaciais incluem as pesquisas de Husain & Natarajan (2020), Khan et al. (2022) e Ahmad et al. (2019) [14] [15] [16].

Esses estudos evidenciam que a integração de CNNs e LSTMs é particularmente eficaz para dados que apresentam informações espaciais e temporais, justificando, portanto, sua relevância para a análise de sinais de EEG no contexto desta pesquisa.

## 3 Métodos

Este estudo realizou a classificação binária de sinais de EEG utilizando duas abordagens distintas: uma abordagem auto-supervisionada e uma abordagem supervisionada. Na abordagem auto-supervisionada, uma cabeça de predição Perceptron Multicamadas (MLP) foi aco-  
plada a um *backbone* previamente pré-treinado por meio do algoritmo de *Contrastive Predic-*

*tive Coding* (CPC). Por outro lado, a abordagem supervisionada baseou-se na utilização de uma arquitetura híbrida combinando redes Convolucionais e *Long short-term memory* (CNN-LSTM).

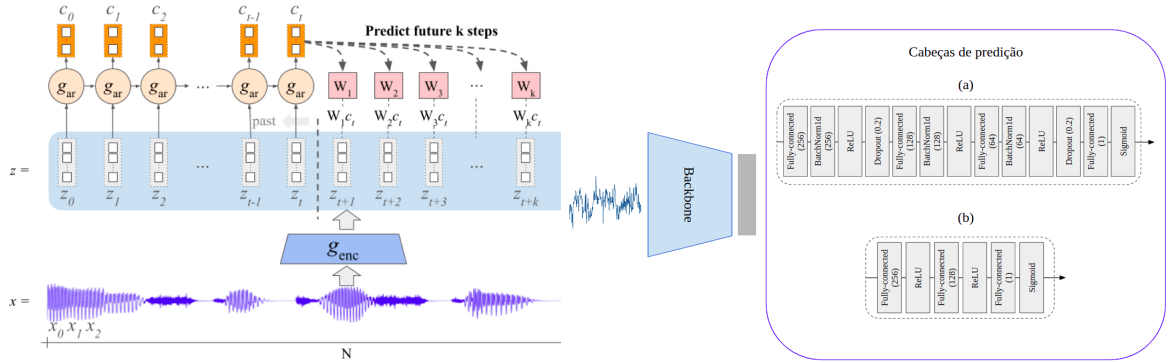
### 3.1 Abordagem auto-supervisionada

#### 3.1.1 Backbone

Considerando que CPC já foi aplicado ao domínio de séries temporais, decidimos explorar sua aplicabilidade em dados médicos, apesar das diferenças significativas entre os domínios e as tarefas alvo.

O algoritmo utilizado foi o proposto pelos autores do CPC para *HAR*<sup>1</sup>, com algumas modificações, especialmente para adequação à entrada do *encoder*. No novo contexto, o modelo precisa suportar 19 canais (eletrodos), enquanto no caso original eram apenas 6 (3 de acelerômetro e 3 de giroscópio).

Conforme ilustrado na Figura 1a, o pré-treinamento do *backbone*, ou extrator de características, é realizado por meio de uma tarefa de pretexto que visa otimizar a capacidade do modelo de capturar informações temporais relevantes. O *backbone* é composto por uma combinação de *genc*, que consiste em três blocos convolucionais compostos por *Conv1d*, ReLU e *dropout*, e *gar*, implementado como uma unidade recorrente fechada (*GRU*). A tarefa pretexto envolve a seleção aleatória de um ponto no tempo dentro de uma janela do sinal, dividindo-a em duas partes: passado e futuro. A partir das representações codificadas do segmento passado, o modelo busca prever a parte futura da janela. O objetivo central é maximizar a informação mútua entre o vetor de contexto, derivado das representações do passado, e a amostra futura que se deseja prever.



(a) Técnica de aprendizado auto-supervisionado *Contrastive Predictive Coding* (CPC).

(b) Diagrama do modelo final de classificação, composto pelo *backbone* e duas cabeças de predição: (a) uma MLP com maior complexidade e (b) uma MLP de menor complexidade.

Figura 1: Diagramas do *backbone* e do modelo final.

#### 3.1.2 Cabeças de predição

Após o pré-treinamento do *backbone*, este foi incorporado ao modelo final projetado para a tarefa de classificação binária dos sinais de EEG. A Figura 1b apresenta a arquitetura proposta, que é composta por duas cabeças de predição distintas, denominadas (a) e (b). Ambas as cabeças são implementadas como Perceptrons Multicamadas (MLPs) acoplados ao *backbone*.

<sup>1</sup><https://github.com/harkash/contrastive-predictive-coding-for-har>

A arquitetura da cabeça (a) é composta por uma camada totalmente conectada com 256 unidades, seguida por normalização de *batch*, ativação ReLU e regularização por *dropout* com taxa de 0,2. Esse padrão é replicado para um bloco de 128 unidades, exceto pela ausência de *dropout*. Posteriormente, a saída é processada por uma camada totalmente conectada com 64 unidades, novamente acompanhada de normalização de *batch* e ativação ReLU. Por fim, após um segundo *dropout*, a saída é gerada por uma camada totalmente conectada com ativação sigmoidal, responsável por produzir a predição final.

Em contraste, a cabeça (b) apresenta uma arquitetura mais simplificada, consistindo em duas camadas totalmente conectadas com 256 e 128 unidades, respectivamente, cada uma seguida por uma ativação ReLU. A predição final é produzida por uma última camada totalmente conectada de um único neurônio com ativação sigmoidal.

## 3.2 Abordagem supervisionada

### 3.2.1 Convolução 1D

As operações de convolução (ou correlação cruzada, propriamente dito) são operações fundamentais em redes convolucionais que permitem a extração de padrões locais de dados, como imagens ou séries temporais [17].

A operação é definida por

$$s(t) = (x * w)(t) = \sum_{\tau=0}^{k-1} x(t + \tau)w(\tau),$$

sendo  $x$  o vetor de entrada (como uma série temporal),  $w$  o vetor de pesos (filtro ou *kernel*, é a posição no vetor de saída  $s(t)$   $k$  sendo o tamanho do *kernel*  $w$ .

### 3.2.2 LSTM

A rede neural recorrente LSTM, introduzido por Hochreiter (1997) [18], para resolver problemas envolvendo desaparecimento de gradientes utiliza o recuso de portas lógicas, que com determinado treinamento, consegue configurá-las para permitir a passagem de informação com diferentes variações temporais, do curto ao longo prazo) [19] [20].

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f),$$

onde  $W_f$  são os pesos,  $b_f$  é o *bias*,  $h_{t-1}$  é o estado oculto anterior e  $x_t$  é a entrada atual.

### 3.2.3 CNN+LSTM

Considerando o exposto, a abordagem supervisionada consiste em combinar os dois tipos de redes para classificação binária dos sinais de EEG. Com isso, a arquitetura composta por camadas convolucionais 1D, camada densa e LSTM serão utilizadas para a tarefa citada.

A arquitetura proposta combina componentes de redes convolucionais (CNNs) e redes recorrentes (LSTMs), otimizando para a tarefa de classificação binária com um modelo projetado para capturar características temporais e espaciais. O modelo recebe sequências temporais de `n_timesteps` e `n_features` e passa por **duas camadas Conv1D** com dois *kernels* de tamanho 3. As camadas de ativação utilizadas são do tipo **ReLU**, e também há a presença do parâmetro de regularização **l2**. Faz-se presente também um `MaxPooling1D` e uma configuração de *dropout*. Em consonância, a camada recorrente de **LSTM** captura as dependências temporais. Por fim, adiciona-se a camada de ativação **tanh** na estrutura.

## 4 Experimentos

### 4.1 Conjunto de dados

O conjunto de dados utilizado neste trabalho consiste em gravações de EEG [6] de 88 indivíduos, divididos em três grupos: 36 pacientes com Alzheimer (AD), 23 pacientes com Demência Frontotemporal (FTD) e 29 indivíduos saudáveis (CN). As gravações foram realizadas utilizando 19 eletrodos posicionados no couro cabeludo e dois eletrodos de referência, com uma taxa de amostragem inicial de 500 Hz. A duração das gravações variou conforme o grupo: aproximadamente 13,5 minutos para pacientes AD (mínimo: 5,1 minutos; máximo: 21,3 minutos), 12 minutos para pacientes FTD (mínimo: 7,9 minutos; máximo: 16,9 minutos) e 13,8 minutos para indivíduos CN (mínimo: 12,5 minutos; máximo: 16,5 minutos).

Os dados originais de EEG foram disponibilizados no formato .set, o que pode dificultar a manipulação desses dados com bibliotecas comuns de aprendizado de máquina na linguagem Python, como o numpy. Com base neste problema, este trabalho aplicou o pré-processamento descrito por MedFormer<sup>2</sup> [21], o qual consistiu em aplicar um filtro passa-faixa entre 0,5 e 45 Hz em cada gravação, reamostrar os sinais para 256 Hz, segmentá-los em janelas não sobrepostas de 1 segundo (256 timestamps por janela) e salvá-los no formato .npy. Além disso, amostras com menos de 1 segundo foram descartadas. Esse processo resultou em um total de 69.752 amostras.

Para garantir a generalização e evitar vazamento de dados entre os conjuntos de treino, teste e validação, a divisão dos dados foi realizada por paciente, com 60% dos indivíduos designados para treino, 20% para validação e 20% para teste. Essa estratégia assegura que amostras do mesmo indivíduo não estejam simultaneamente em diferentes conjuntos.

### 4.2 Configuração experimental

Para comparar os modelos propostos neste trabalho com o *benchmark* implementado pelo artigo do conjunto de dados original [5], foram treinados dois classificadores para cada abordagem, sendo um para a classificação AD/CN e outro para a classificação FTD/CN.

#### 4.2.1 Abordagem auto-supervisionada

O treinamento do *backbone* foi realizado em uma GPU RTX 5000, utilizando um tamanho de *batch* de 64, uma taxa de aprendizado de  $5e^{-4}$  ao longo de 150 épocas. Após o pré-treinamento, foram experimentadas duas configurações distintas de cabeças de predição acopladas ao *backbone*. O treinamento do modelo final foi conduzido em uma GPU NVIDIA T4, disponibilizada pelo Google Colab, com tamanho de *batch* de 64. Foram avaliadas duas estratégias distintas para o treinamento: congelamento dos parâmetros do *backbone* e ajuste fino (*fine-tuning*). Na estratégia de congelamento, os parâmetros do *backbone* permaneceram fixos, e a taxa de aprendizado foi configurada para  $1e^{-5}$ . Já na estratégia de ajuste fino, os parâmetros do *backbone* foram atualizados durante o treinamento, utilizando uma taxa de aprendizado reduzida de  $1e^{-6}$ .

#### 4.2.2 Abordagem supervisionada

O treinamento do modelo CNN+LSTM envolveu de uma CPU 12th Gen Intel(R) Core(TM) i5-12500H 2.50 GHz. Foram fixadas 20 épocas de treino, mas também nesse caso o modelo acionou o *early-stopping*. Os learning rates utilizados foram de 0.0003 e 0.0001 para o classificador AD/CN e FTD/CN, respectivamente. O *batch size* utilizado foi de 128.

<sup>2</sup>[https://github.com/DL4mHealth/Medformer/blob/main/data\\_preprocessing/ADFTD\\_preprocessing.ipynb](https://github.com/DL4mHealth/Medformer/blob/main/data_preprocessing/ADFTD_preprocessing.ipynb)

Na fase de otimização de hiperparâmetros utilizou-se experimentos com 20 trials (10 épocas de treino para cada trial). O experimento envolveu a otimização de hiperparâmetros com o **Optuna**, um framework de otimização inteligente que otimiza e torna dinâmica a busca de hiperparâmetros de modelos de *machine learning* [22]. A métrica que se visou otimizar foi a *roc-auc-score*.

Parâmetros	Interval	Type
filters_1	{16, 32, 64}	Categorical
filters_2	{32, 64, 128}	Categorical
lstm_units	{50, 100, 150}	Categorical
dense_units	{16, 32, 64}	Categorical
dropout_rate	[0.1, 0.5]	Float
learning_rate	$10^{-4}$ to $10^{-2}$ (log)	Float (log scale)
l2_reg	$10^{-4}$ to $10^{-3}$ (log)	Float (log scale)

Tabela 1: Malha de hiperparâmetros utilizada para a otimização.

A malha de hiperparâmetros testada para a otimização dos modelos de classificação de ambas enfermidades 1 mostra todos os hiperprâmetros testados em um dado intervalo e o tipo de busca nesse intervalo proposto (*Type*) - se é discreto ou contínuo.

Parâmetros	Alzheimer	Demência
Filters 1	16	64
Filters 2	128	32
LSTM Units	100	100
Dense Units	32	16
Dropout Rate	0.3609	0.5040
Learning Rate	0.0003038	0.0001600
L2 Regularization	0.0051641	0.0032568

Tabela 2: Melhores Parâmetros Encontrados

Os melhores hiperparâmetros encontrados na fase de otimização com o Optuna [22] e utilizados na modelagem final encontra-se na Tabela 2.

### 4.3 Métricas de avaliação

As métricas quantitativas escolhidas para avaliação dos classificadores estão alinhadas com as métricas reportadas no *benchmark* [5], sendo elas a acurácia balanceada (ACC), precisão (P), revocação (R) e f1-score (F1). Além disso, os experimentos da abordagem supervisionada utilizaram a métrica *roc-AUC-score* (AUC).

## 4.4 Resultados

### 4.4.1 Abordagem auto-supervisionada

Os experimentos desta abordagem envolveram o treinamento de quatro modelos para cada tarefa de classificação, utilizando diferentes estratégias de aprendizado. Foram considerados dois cenários principais: treinamento com congelamento dos parâmetros do *backbone* e cabeças de predição 1b(a) e 1b(b), e treinamento com ajuste fino dos parâmetros do *backbone* e cabeças de predição 1b(a) e 1b(b).

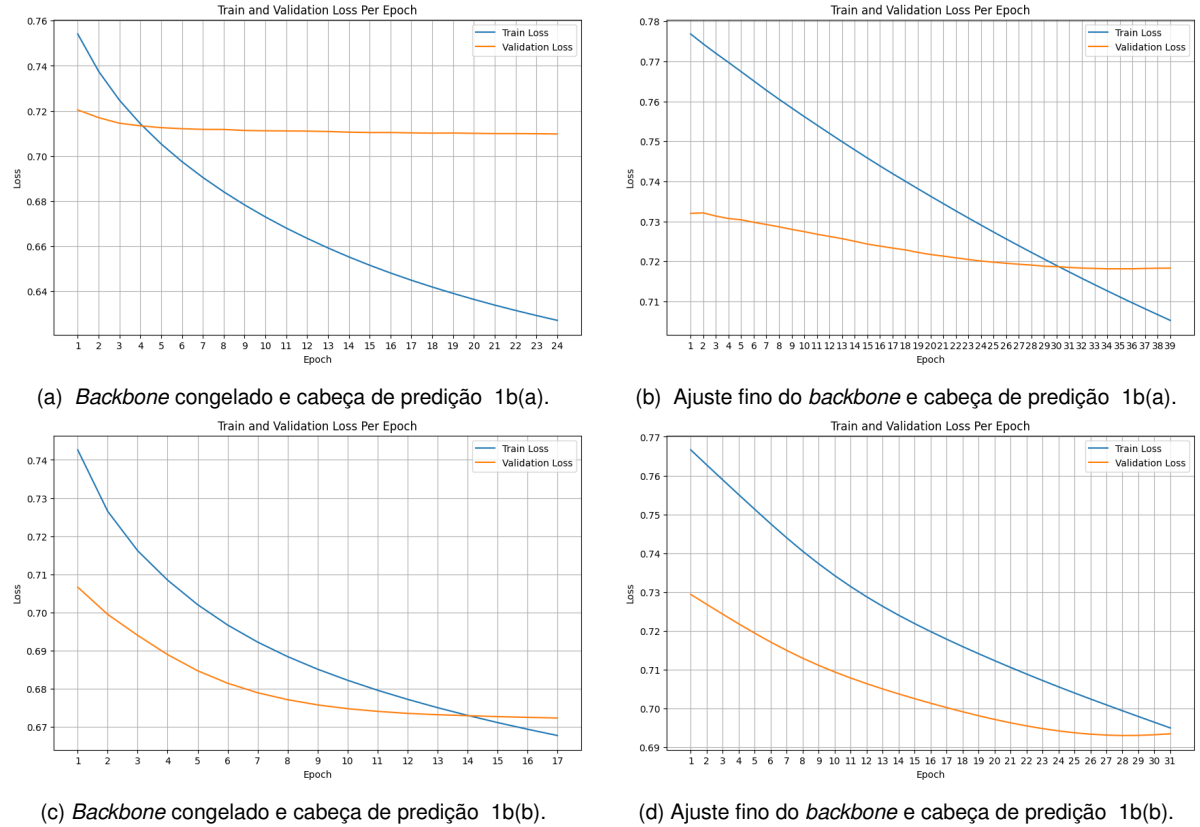


Figura 2: Gráficos das curvas de perda de treino e validação dos classificadores AD/CN.

A Figura 5a demonstra que, em poucas épocas, a curva de validação dos classificadores AD/CN estabiliza rapidamente, enquanto a curva de treino continua a evoluir, comportamento mais evidente nos experimentos com o congelamento dos parâmetros do *backbone* (Figuras 2a e 2c). Um comportamento semelhante pode ser observado nos gráficos das curvas de perda do classificador FTD/CN 3, porém a estabilização da curva de validação é muito mais precoce do que a observada no treinamento do classificador AD/CN. Para evitar o sobreajuste, foi implementado o critério de *early stopping*, interrompendo o treinamento assim que o desempenho na validação não apresentasse melhorias significativas.

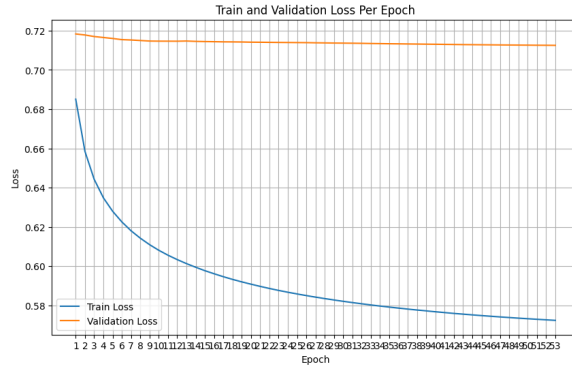
Após o treinamento, uma avaliação preliminar foi realizada no conjunto de validação, uma vez que, devido ao estágio de experimentação e escolha do melhor modelo, o conjunto de testes não foi utilizado para evitar qualquer viés na adaptação do modelo a esse conjunto específico. Os resultados para a tarefa de classificação AD/CN estão apresentados na Tabela 3, enquanto os resultados para a classificação FTD/CN são exibidos na Tabela 4.

Tabela 3: Resultados quantitativos da classificação AD/CN no conjunto de validação.

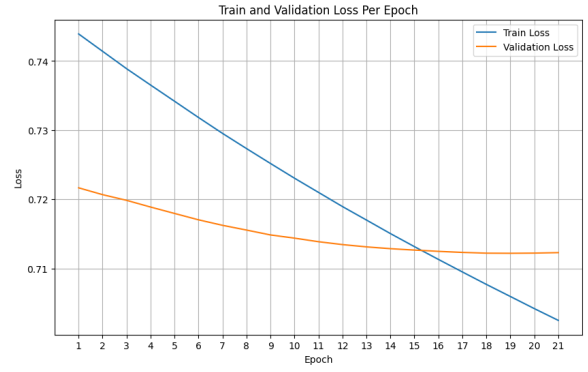
AD/CN	ACC	P	R	F1
<i>Backbone congelado e cabeça de predição 1b(a)</i>	62%	62%	62%	62%
<i>Ajuste fino do backbone e cabeça de predição 1b(a)</i>	61%	61%	61%	61%
<i>Backbone congelado e cabeça de predição 1b(b)</i>	<b>63%</b>	<b>65%</b>	<b>63%</b>	<b>63%</b>
<i>Ajuste fino do backbone e cabeça de predição 1b(b)</i>	56%	57%	56%	56%

Com base nos resultados, observa-se que o melhor modelo para ambas as tarefas foi o com *backbone* congelado e cabeça de predição 1b(b). Para a classificação AD/CN este modelo alcançou 63% de acurácia balanceada, 65% de precisão, 63% de revocação e 63%

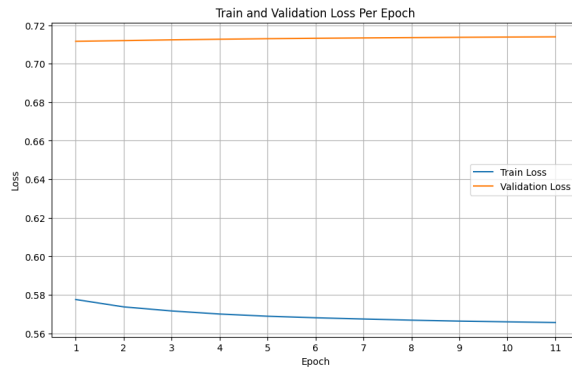




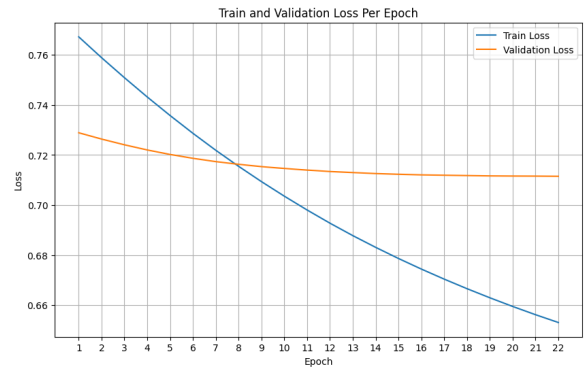
(a) *Backbone* congelado e cabeça de predição 1b(a).



(b) Ajuste fino do *backbone* e cabeça de predição 1b(a).



(c) *Backbone* congelado e cabeça de predição 1b(b).



(d) Ajuste fino do *backbone* e cabeça de predição 1b(b).

Figura 3: Gráficos das curvas de perda de treino e validação dos classificadores FTD/CN.

de f1-score. Já para a classificação FTD/CN o modelo alcançou 64% de acurácia balanceada, 67% de precisão, 64% de revocação e 64% de f1-score.

Tabela 4: Resultados quantitativos da classificação FTD/CN no conjunto de validação.

FTD/CN	ACC	P	R	F1
<i>Backbone</i> congelado e cabeça de predição 1b(a)	61%	61%	61%	61%
Ajuste fino do <i>backbone</i> e cabeça de predição 1b(a)	56%	56%	56%	56%
<i>Backbone</i> congelado e cabeça de predição 1b(b)	<b>64%</b>	<b>67%</b>	<b>64%</b>	<b>64%</b>
Ajuste fino do <i>backbone</i> e cabeça de predição 1b(b)	61%	61%	61%	61%

#### 4.4.2 Análise das representações aprendidas

Para visualizar as representações aprendidas pelo *backbone*, foram geradas visualizações com a técnica *t-distributed Stochastic Neighbor Embedding* (t-SNE) dos dados brutos e das características extraídas pelo modelo (Figura 4). Os dados utilizados para essa análise incluem informações de 51 pacientes, distribuídos em três classes: 21 pacientes da classe AD, 13 pacientes da classe FTD e 17 pacientes da classe CN. Essa abordagem permite uma compreensão mais intuitiva da separação e estrutura das diferentes classes no espaço de características, fornecendo uma melhor compreensão sobre a capacidade do modelo em distinguir as classes a partir dos dados brutos e das representações aprendidas.

As visualizações geradas levam a uma primeira conclusão de que o *backbone* é capaz de promover uma boa separabilidade entre as classes. No entanto, ao analisar as representações

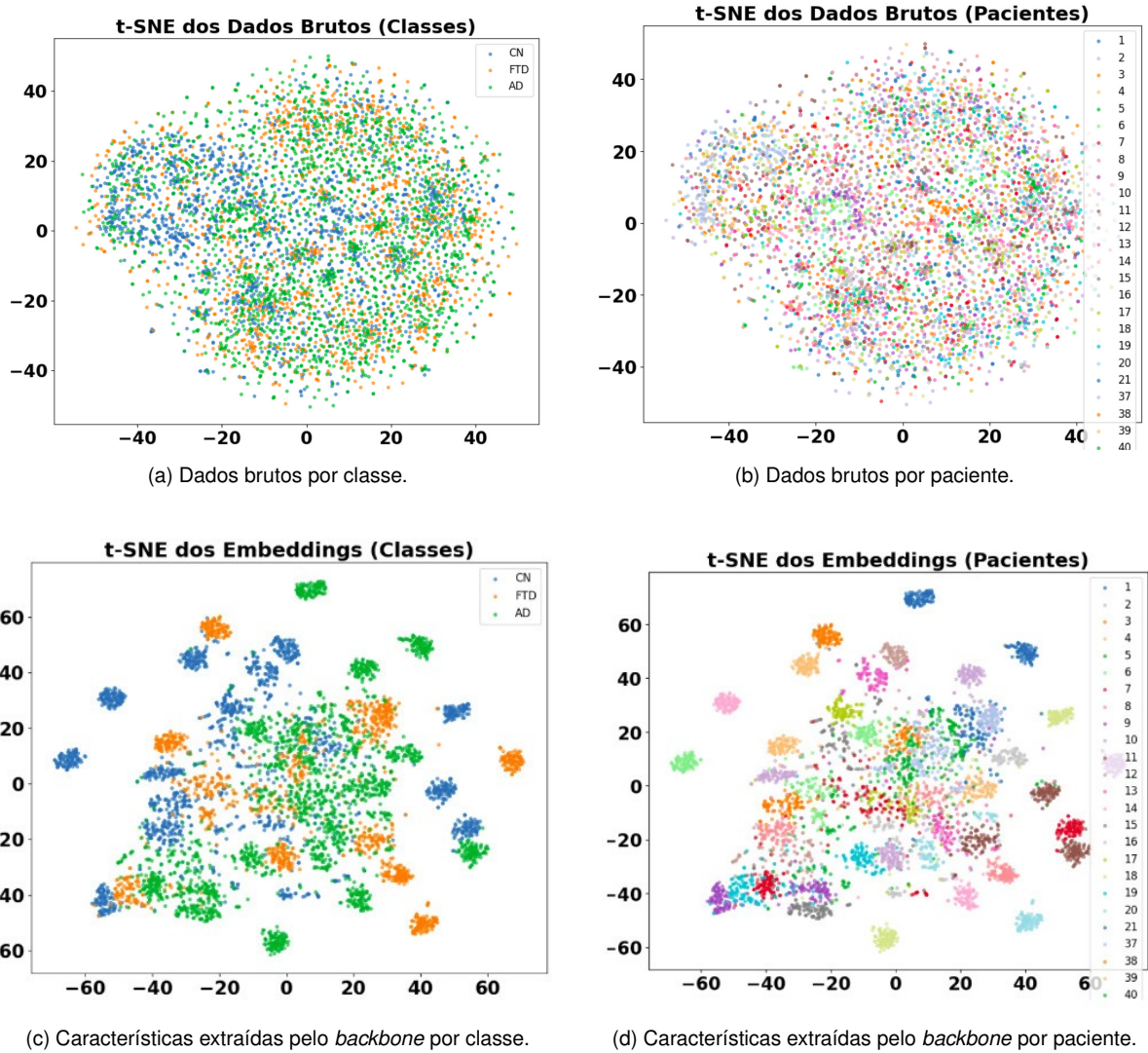


Figura 4: Visualização dos dados brutos e características extraídas pelo *backbone* de 51 pacientes utilizando a técnica t-SNE.

aprendidas por classes e por pacientes, observa-se que o *backbone* aprendeu as características entre pacientes, em vez de focar na distinção entre classes. Esse comportamento é totalmente prejudicial ao desempenho da classificação, pois, ao realizar a predição em dados de pacientes nunca vistos, o modelo tende a falhar na identificação correta das classes, uma vez que as características extraídas não são generalizáveis. Assim, faz-se necessário o ajuste da abordagem de aprendizado para que o *backbone* priorize a captura de padrões relevantes entre as classes, independentemente da identidade dos pacientes, aumentando a capacidade de generalização do modelo.

#### 4.4.3 Abordagem supervisionada

Os resultados finais da arquitetura envolvendo redes CNN+LSTM mostram que o treinamento enfrentou dificuldades significativas com *overfitting*. Apesar de utilizarmos dropout nas camadas convolucionais do modelo (36% para AD/CN e 50% para FTD/CN) e adicionarmos regularização L2, o modelo ainda apresentou problemas. O mecanismo de *early stopping* foi ativado logo nas primeiras épocas de treinamento, indicando que o modelo parou de melhorar rapidamente 5a 5b. Isso sugere que, mesmo com essas técnicas de regularização, o modelo não conseguiu generalizar bem.

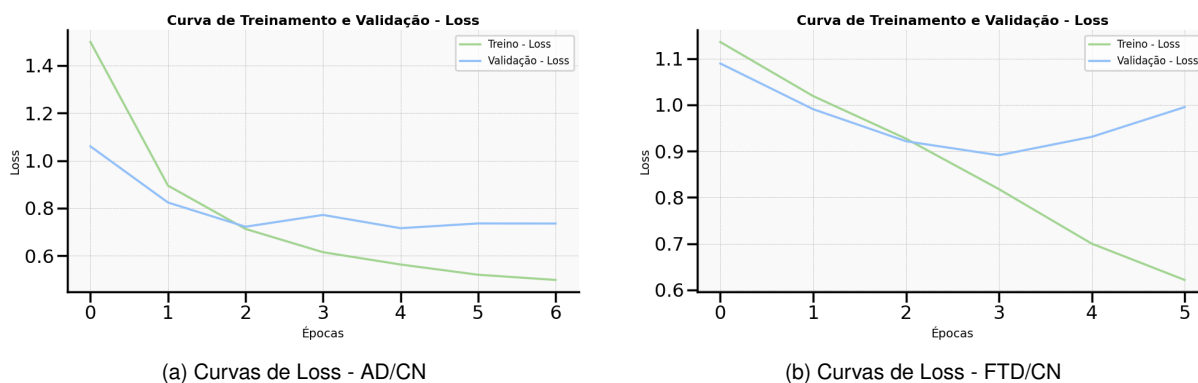


Figura 5: Comparação das curvas de Loss entre AD/CN e FTD/CN.

Após o treinamento, a escolha do *threshold* de decisão foi algo que se buscou cautela, uma vez que se tratando de um problema médico buscou-se favorecer a métrica de revocação em detrimento de precisão.

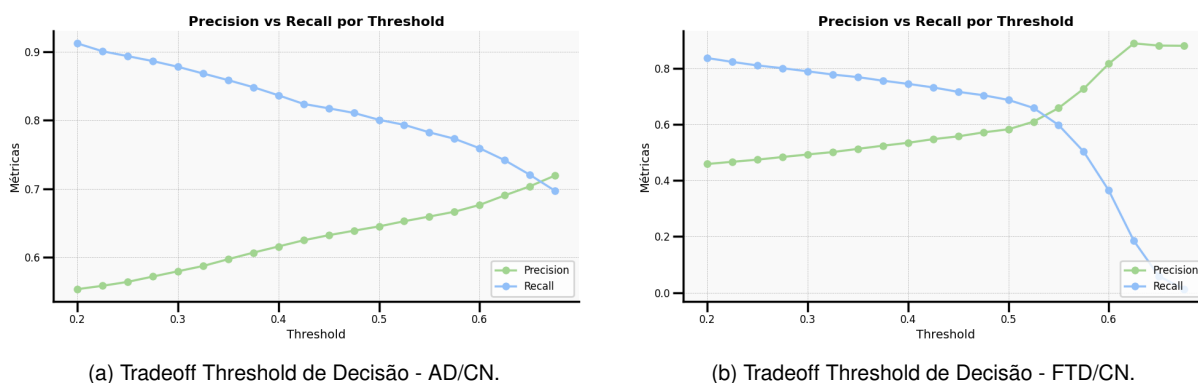


Figura 6: Comparação dos tradeoffs de threshold de decisão entre AD/CN e FTD/CN.

De acordo com 6a, buscou-se o *threshold* que conseguisse favorecer a revocação, isto é, minimizar o máximo possível a ocorrência de falsos negativos mesmo que o modelo incorresse em maior quantidade de falsos positivos. Com isso, o limiar escolhido para AD/CN foi de 0.40 e FTD/CN 6b foi de 0.35.

Tabela 5: Comparação das métricas de validação e teste para Alzheimer e Demência.

Classe	Validação (P/R)	Teste (P/R)
<b>AD/CN (0)</b>	65% / 32%	74% / 35%
<b>AD/CN (1)</b>	60% / 86%	65% / 90%
<b>FTD/CN (0)</b>	65% / 42%	70% / 43%
<b>FTD/CN (1)</b>	53% / 75%	44% / 71%

Vale notar que observando as métricas da tabela 5- que descrever as métricas de *precision* e *recall* para amostras apenas da classe 1 (enfermidade) e classe 0 (saúdável) de forma individual foi possível conseguir resultados mais elevados de *recall* 86% para AD/CN e 75% para FTD/CN no conjunto de validação. Já para o conjunto de teste constatado um *recall* de 90% para AD/CN e 71% para FTD/CN. Acredita-se que os esforços para mitigar a ocorrência de falsos negativos contribuiu sobremaneira para esses resultados.

Tabela 6: Resultados quantitativos da classificação AD/CN no conjunto de validação.

AD/CN	ACC	P	R	F1	AUC
CNN + LSTM	61%	63%	59%	57%	73%

Tabela 7: Resultados quantitativos da classificação FTD/CN no conjunto de validação.

FTD/CN	ACC	P	R	F1	AUC
CNN + LSTM	58%	59%	59%	57%	67%

As tabelas 6 e 7 apresentam os resultados quantitativos da classificação para o conjunto de validação. A partir da avaliação das métricas percebe-se que o modelo alcançou revocação de 59% e *f1-score* de 57% para ambas as enfermidades. No caso de AD/CN, a acurácia balanceada alcançada foi de 61%, precisão de 63% e AUC de 73%. Já para FTD/CN o modelo alcançou 58% de acurácia balanceada, 59% de precisão e 67% de AUC.

#### 4.4.4 Comparação com o *benchmark*

Após a seleção do modelo mais eficaz para ambas as abordagens, com base no desempenho no conjunto de validação, foi realizada uma avaliação comparativa final com o *benchmark* no conjunto de teste. Para isso, foram selecionados os melhores modelos reportados para cada tarefa no *benchmark*, sendo o Random Forests para a classificação AD/CN e o MLP para a classificação FTD/CN.

Tabela 8: Resultados quantitativos da classificação AD/CN no conjunto de teste.

AD/CN	ACC	P	R	F1
Random Forests ( <i>Benchmark</i> )	<b>81%</b>	57%	<b>78%</b>	<b>66%</b>
Modelo SSL	63%	<b>65%</b>	63%	63%
CNN + LSTM	61%	63%	59%	57%

A Tabela 8 apresenta os resultados do *benchmark*, do melhor modelo da abordagem auto-supervisionada e da rede CNN+LSTM para a classificação AD/CN. De forma análoga, a Tabela 9 exibe os resultados obtidos pelos três modelos na classificação FTD/CN. Em ambas as tarefas, o *benchmark* apresentou o melhor desempenho geral, destacando-se frente às demais abordagens.

Tabela 9: Resultados quantitativos da classificação FTD/CN no conjunto de teste.

FTD/CN	ACC	P	R	F1
MLP ( <i>Benchmark</i> )	<b>73%</b>	<b>87%</b>	63%	<b>73%</b>
Modelo SSL	64%	67%	<b>64%</b>	64%
CNN + LSTM	58%	59%	59%	57%

## 5 Conclusão e Trabalhos Futuros

### 5.1 Conclusão

Este trabalho nos proporcionou a oportunidade de abordar um problema relevante na área médica, enfrentando o desafio de aplicar técnicas de aprendizado de máquina com eficácia. Uma das lições aprendidas foi que o sucesso neste campo frequentemente exige experimentação cuidadosa, orientada por intuições baseadas em trabalhos prévios e evidências práticas. Apesar dos esforços, as métricas obtidas neste estudo não alcançaram níveis satisfatórios para aplicações reais, indicando a necessidade de mais experimentação e refinamento.

A visualização dos *embeddings* no espaço latente, gerado pelo CPC com o auxílio do TSNE, revelou que o modelo conseguiu identificar padrões. Contudo, observamos que o aprendizado separou os pacientes em vez de distinguir as doenças. Essa característica pode ser atribuída ao fato de que cada paciente representa um domínio distinto, já que as condições específicas do couro cabeludo durante a coleta dos dados de EEG variam significativamente entre os indivíduos. Assim, o modelo, durante a tarefa pretexto do CPC, aprendeu a diferenciar os domínios individuais em vez de focar nos padrões relacionados às doenças.

No caso da abordagem supervisionada conseguimos explorar a diversidade dos mecanismos de busca de hiperprâmetros e variação do *threshold* de decisão. Essas etapas proporcionaram uma escolha entre o modelo focar sua escolha mais na classe doente do que a saudável, por isso alcançou-se um recall considerável para a classe 1 (em que a prioridade foi a minimização dos falsos negativos). Mesmo com todos os cuidados sobre essa etapa supervisionada, terminou-se enfrentando problemas de *overfitting* apesar de termos utilizado múltiplas estratégias de regularização.

Para trabalhos futuros, sugerimos explorar algumas direções promissoras. Primeiro, aumentar a janela dos dados, atualmente limitada a 256 amostras (equivalente a 1 segundo), o que pode ter sido insuficiente para capturar informações relevantes. Além disso, seria interessante investigar outras tarefas pretexto, mais alinhadas com o domínio de detecção de anomalias. Alternativas que não dependam da predição do tempo futuro, como no CPC, podem revelar-se mais adequadas para os desafios apresentados neste contexto específico. Outras variações de arquiteturas para esses tipos de dados poderiam ser exploradas.

### 5.2 Trabalhos Futuros e Melhorias

#### 5.2.1 Mudar o tamanho da janela de dados de 256 para 512

O pré-processamento realizado pelo trabalho MedFormer [21] segmentou os dados em janelas com 256 pontos de dados, equivalentes a um segundo de coleta. Eventualmente, os padrões que buscamos identificar podem não se manifestar em janelas de dados tão pequenas, tornando interessante o estudo de outros tamanhos de janelas.

#### 5.2.2 Permitir a sobreposição de dados nas sequências geradas

O pré-processamento realizado pelo trabalho MedFormer [21] também garantiu a não sobreposição de dados entre as sequências formadas em cada janela. Contudo, análises clássicas de séries temporais tipicamente consideram alguma sobreposição de dados, o que pode enriquecer o conjunto de dados para a modelagem.

### 5.2.3 Analisar quais canais podem ser mais importantes para a detecção de uma doença

O artigo [5], referência para o conjunto de dados utilizado, cita que existem canais mais apropriados para a detecção de doenças específicas e que há transformações que podem ajudar nesse sentido. Dessa forma, um passo adicional no pré-processamento dos dados pode fazer a diferença e contribuir para a diferenciação entre as classes.

### 5.2.4 Revisitar questões de balanceamento

Em relação ao número de pacientes, a base de dados está relativamente balanceada. Contudo, existem outros aspectos que podem ser explorados mais a fundo, como o grau da doença, o sexo do paciente, a idade, o número de sequências geradas pelos dados de cada paciente, entre outras variáveis.

### 5.2.5 Explorar diferentes tarefas pretexto

No caso da abordagem auto-supervisionada, seria interessante investigar outras tarefas pretexto mais alinhadas com o domínio de detecção de anomalias. Alternativas que não dependam da predição do tempo futuro, como no CPC, podem revelar-se mais adequadas para os desafios apresentados neste contexto específico.

## 6 Considerações Éticas

Este trabalho utiliza dados de saúde humana, o que exige atenção rigorosa às questões éticas e de privacidade. Os dados, originalmente públicos e anonimizados, seguem o Regulamento Geral sobre a Proteção de Dados (GDPR), assegurando a proteção contra qualquer identificação dos participantes. A coleta inicial foi aprovada pelo Comitê de Ética do Hospital Universitário AHEPA, conforme o protocolo 142/12-04-2023, e conduzida segundo os princípios da Declaração de Helsinque.

Destaca-se que ferramentas de inteligência artificial (IA) têm o potencial de apoiar decisões médicas, incluindo o diagnóstico. No entanto, é fundamental ressaltar que o mau uso dessas ferramentas pode trazer consequências prejudiciais aos pacientes, especialmente quando diagnósticos são realizados sem a devida avaliação de um especialista. Assim, reforça-se que os modelos preditivos não substituem a análise clínica completa e o julgamento humano especializado.

Os modelos desenvolvidos neste trabalho ainda estão aquém do desejável para aplicações em cenários críticos reais. Nesse sentido, é imprescindível refletir sobre os custos éticos associados aos erros de classificação. Por exemplo, qual é o impacto de diagnosticar falsamente um paciente saudável como doente? Ou, de forma ainda mais preocupante, qual é o custo ético de classificar como saudável um paciente que está no estágio inicial de uma doença potencialmente controlável com um diagnóstico precoce? Estas questões ressaltam a necessidade de cautela no uso de sistemas automatizados de apoio ao diagnóstico médico e reforçam a importância de estudos futuros para mitigar tais riscos.

## Referências

- [1] P. Attia and B. Gifford, *Outlive: A arte e a ciência de viver mais e melhor*. Intrínseca, August 2023.

- [2] Secretaria Municipal da Saúde de Curitiba, “Saúde já vai diminuir espera por exames e as filas de consultas de especialidades,” 2024, acessado em: 12 dez. 2024. [Online]. Available: <https://saude.curitiba.pr.gov.br/noticias/23-noticias-principal/618-saude-ja-vai-diminuir-espera-por-exames-e-as-filas-de-consultas-de-especialidades>
- [3] D. P. Subha, P. K. Joseph, U. R. Acharya, and K. Choo, “Eeg signal analysis: A survey,” *Journal of Medical Systems*, vol. 34, pp. 195–212, 2010. [Online]. Available: <https://doi.org/10.1007/s10916-008-9231-z>
- [4] A. Miltiadous, K. D. Tzimourta, N. Giannakeas, M. G. Tsipouras, T. Afrantou, P. Ioannidis, and A. T. Tzallas, “Alzheimer’s disease and frontotemporal dementia: A robust classification method of eeg signals and a comparison of validation methods,” *Diagnostics*, vol. 11, no. 8, 2021. [Online]. Available: <https://www.mdpi.com/2075-4418/11/8/1437>
- [5] A. Miltiadous, K. D. Tzimourta, T. Afrantou, P. Ioannidis, N. Grigoriadis, D. G. Tsalikakis, P. Angelidis, M. G. Tsipouras, E. Glavas, N. Giannakeas, and A. T. Tzallas, “A dataset of scalp eeg recordings of alzheimer’s disease, frontotemporal dementia and healthy subjects from routine eeg,” *Data*, vol. 8, no. 6, 2023. [Online]. Available: <https://www.mdpi.com/2306-5729/8/6/95>
- [6] —, “A dataset of EEG recordings from Alzheimer’s disease, Frontotemporal dementia and Healthy subjects,” 2023. [Online]. Available: <https://openneuro.org/datasets/ds004504/versions/1.0.7>
- [7] R. Balestrieri, M. Ibrahim, V. Sobal, A. Morcos, S. Shekhar, T. Goldstein, F. Bordes, A. Bardes, G. Mialon, Y. Tian *et al.*, “A cookbook of self-supervised learning,” *arXiv:2304.12210*, 2023.
- [8] L. Ericsson, H. Gouk, C. C. Loy, and T. M. Hospedales, “Self-supervised representation learning: Introduction, advances, and challenges,” *IEEE Signal Processing Magazine*, vol. 39, no. 3, 2022.
- [9] A. v. d. Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” *arXiv:1807.03748*, 2018.
- [10] H. Haresamudram, I. Essa, and T. Plötz, “Contrastive predictive coding for human activity recognition,” *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 5, no. 2, 2021.
- [11] X. Zhang, Z. Zhao, T. Tsiligkaridis, and M. Zitnik, “Self-supervised contrastive pre-training for time series via time-frequency consistency,” *arXiv preprint arXiv:2206.08496*, 2022, version 3, last revised 15 Oct 2022. [Online]. Available: <https://arxiv.org/abs/2206.08496>
- [12] J. Zhao, X. Mao, and L. Chen, “Speech emotion recognition using deep 1d 2d cnn lstm networks,” *Biomed. Signal Process. Control.*, vol. 47, pp. 312–323, 2019.
- [13] R. Mutegeki and D. S. Han, “A cnn-lstm approach to human activity recognition,” in *2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, 2020, pp. 362–366.
- [14] S. Husain and B. S. Natarajan, “Enhanced cnn-lstm approach for human activity recognition,” vol. 9, pp. 199–205, 2020.
- [15] I. Khan, S. Afzal, and J.-W. Lee, “Human activity recognition via hybrid deep learning based model,” *Sensors (Basel, Switzerland)*, vol. 22, 2022.

- [16] W. Ahmad, M. Kazmi, and H. Ali, "Human activity recognition using multi-head cnn followed by lstm," *2019 15th International Conference on Emerging Technologies (ICET)*, pp. 1–6, 2019.
- [17] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The handbook of brain theory and neural networks*. MIT Press, 1995, pp. 255–258.
- [18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [19] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.
- [20] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv preprint arXiv:1402.1128*, 2014.
- [21] Y. Wang, N. Huang, T. Li, Y. Yan, and X. Zhang, "Medformer: A multi-granularity patching transformer for medical time-series classification," *arXiv preprint arXiv:2405.19363*, 2024.
- [22] Optuna Development Team, *Optuna: A hyperparameter optimization framework*, Optuna Development Team, accessed: 2024-12-12. [Online]. Available: <https://optuna.readthedocs.io/en/stable/>