

Modelo Baseado em Transformer para Auxiliar no Diagnóstico de Esclerose Lateral Amiotrófica a partir da Marcha

Giordana de Farias F. B. Bucci¹, Ana Luísa de Bastos Chagas¹, Pedro L. S. Lobo¹
Hugo A. D. do Nascimento¹, Fabrizzio Soares¹, Juliana Felix^{1,2}

¹Instituto de Informática, Universidade Federal de Goiás (UFG)

²Escola Politécnica e de Artes, Pontifícia Universidade Católica de Goiás (PUC-Goiás)

{giordanabucci, analuisa23, pedro_lemes}@discente.ufg.br
{hadn, fabrizzio, julianafelix}@ufg.br

Abstract. *Amyotrophic Lateral Sclerosis (ALS) causes, among other symptoms, gait instability, is incurable, and involves a long and difficult diagnostic process. Therefore, several studies investigate gait analysis through artificial intelligence models as an alternative to aid in the diagnosis of the disease. This work presents an innovative method for ALS detection using a transformer-based model combined with gait analysis in a binary classification task. The results achieve high accuracy values and indicate a promising alternative for ALS identification.*

Resumo. *Esclerose Lateral Amiotrófica (ELA) causa, entre outros sintomas, instabilidade da marcha, possui natureza incurável e apresenta um longo e difícil processo de diagnóstico. Por isso, diversos estudos investigam a marcha por meio de modelos de inteligência artificial como uma alternativa para auxiliar no diagnóstico dessas doenças. Este trabalho apresenta um método inovador de detecção de ELA utilizando um modelo baseado em transformer associado a análise de marcha em uma tarefa de classificação binária. Os resultados alcançam valores altos de acurácia e indicam uma alternativa promissora de identificação de ELA.*

1. Introdução

Esclerose Lateral Amiotrófica (ELA) é uma doença neurodegenerativa (DND), caracterizada pela degeneração progressiva dos neurônios, o que afeta diretamente o sistema nervoso e impacta os movimentos corporais [Berman and Bayati, 2018]. Os sinais apresentados e os sintomas causados por essa doença, apesar de variar entre indivíduos, incluem dificuldade na marcha, tropeços, quedas e fraqueza muscular [Goldfarb and Simon, 1984]. Com o avanço da doença, funções como mastigação, fala e respiração são comprometidas [Masrori and Van Damme, 2020].

Não há cura para ELA [Heemels, 2016] e não há, atualmente, um teste definitivo da doença [Erkkinen et al., 2018]. Diante disso, o diagnóstico é um processo complexo e puramente clínico que baseia-se na observação, na análise e no acompanhamento médico da progressão dos sintomas ao longo do tempo [Mayeux, 2003; Hardiman et al., 2011], o que pode levar meses ou até mesmo anos para ser concluída. Isso ocasiona em um diagnóstico tardio da ELA, de modo que retarda o início do tratamento e agrava os sintomas, comprometendo a qualidade de vida dos acometidos por essa doença.

Diante desse cenário, a análise de marcha é um método alternativo atrativo para auxiliar no diagnóstico precoce de doenças neurodegenerativas, inclusive Esclerose Lateral Amiotrófica [Hausdorff et al., 2000]. Na literatura, o estudo da marcha para identificação de DNDs é comumente encontrado em conjunto com modelos de aprendizado de máquina. Bilgin and Akin [2018] alcançaram 96,55% de acurácia na distinção entre ELA e controles (CO) por meio de uma abordagem que combinou a Transformada Wavelet Discreta (DWT) e o sinal de força composto (CFS) da marcha, validada com o método *Leave-one-out Cross-validation* (LOOCV) em um classificador *Multilayer Perceptron* (MLP).

Elden et al. [2018] utilizaram técnicas de reconstrução de fase com um classificador SVM — kernel de Função de Base Radial (RBF) — e alcançaram 86,21% de acurácia. Gupta et al. [2019] obtiveram uma acurácia de 96,20% ao usar uma Árvore de Decisão com características de autocorrelação. Prabhu et al. [2020] atingiram 96,15% de acurácia com um SVM — kernel RBF — a partir de 28 características utilizando Análise de Quantificação de Recorrência (RQA) e medidas estatísticas. Beyrami and Ghaderyan [2020] reportaram 100% de acurácia a partir da extração de características em janelas de 10 segundos, com um classificador *Sparse NNLS*.

Enquanto esses trabalhos se baseiam em arquiteturas tradicionais de aprendizado de máquina, estudos recentes demonstram o potencial dos modelos baseados em *transformers* para análise de dados sequencias [Agarwal et al., 2020]. O notável avanço refere-se à capacidade do Transformer [Vaswani et al., 2017] em processar e capturar relações contextuais entre os elementos sequenciais, além de “aprender” as dependências à longo prazo utilizando principalmente mecanismos de atenção. Isso é computacionalmente vantajoso, já que o modelo não percorre ou itera sobre os dados, como fazem as redes neurais convolucionais (CNNs), as redes neurais recorrentes (RNNs), as redes *Gated Recurrent Unit* (GRUs) e as redes *Long Short-Term Memory* (LSTMs).

Apesar dessa inovação em análises de séries temporais com *transformer*, apenas alguns trabalhos implementam esse modelo com dados de marcha no domínio de doenças neurodegenerativas, sem foco específico em ELA. Nguyen et al. [2022] abordaram o diagnóstico de Parkinson como uma tarefa de classificação binária, enquanto Sun and Zhang [2022] e Naimi et al. [2023] abordaram a detecção da severidade da doença de Parkinson como uma tarefa de classificação. Já Bucci et al. [2025] abordaram a identificação de algumas doenças neurodegenerativas em uma tarefa de multiclassificação.

Nesse sentido, este artigo propõe um novo método não invasivo para identificação de Esclerose Lateral Amiotrófica e pacientes de controle em uma tarefa de classificação binária. O método utiliza análise de marcha associada a uma arquitetura baseada em *transformer*, a fim de contribuir para o diagnóstico precoce e eficiente de ELA. O restante deste artigo está organizado da seguinte forma: na Seção 2, é detalhada a base matemática que fundamenta o modelo *transformer*; na Seção 3, são explicados os materiais utilizados e os métodos propostos, especificando a base de dados, o pré-processamento dos dados, o modelo implementado e a forma de avaliação dos resultados; na Seção 4, são apresentados os resultados; na Seção 5, é discutido sobre os resultados alcançados; e as conclusões seguem na Seção 6.

2. Fundamentação Teórica

2.1. Mecanismo de Atenção

O mecanismo de atenção é uma técnica que capacita um modelo a focar seletivamente nas partes mais relevantes de uma sequência de entrada, atribuindo pesos de importância dinâmicos a cada elemento. A sua operação se baseia na interação de três componentes vetoriais: a *query* (consulta), a *key* (chave) e o *value* (valor). Matematicamente (Eq. 1), o processo consiste em calcular a similaridade entre as *queries* e as *keys* por meio de um produto interno QK^T , estabilizar o resultado com o fator de escala $1/\sqrt{d_k}$ e, por fim, normalizar esses valores com a função *softmax* (Eq. 2) para obter os pesos finais. Tais pesos determinam a contribuição de cada *value* na composição da saída do mecanismo.

$$Attention(Q, K, V) = softmax \left(\frac{QK^T}{\sqrt{d_k}} \right) V. \quad (1)$$

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}. \quad (2)$$

2.2. Transformer

A arquitetura *Transformer*, proposta por Vaswani et al. [2017], representa um marco no processamento de dados sequenciais. O modelo se baseia em uma estrutura de codificador-decodificador (*encoder-decoder*) e algumas funções adicionais, como *embeddings*, codificação posicional e *softmax*.

O codificador (*encoder*) é responsável por mapear a sequência de entrada para uma representação contínua. É composto por uma pilha de n camadas idênticas, em que cada camada possui duas subcamadas. A primeira é um mecanismo de atenção (*Multi-Head Attention*) e a segunda é uma rede neural totalmente conectada *FeedForward*. Para facilitar o treinamento de redes profundas, cada uma dessas subcamadas é envolvida por uma conexão residual e uma camada de normalização (*Layer Norm*). Todas as camadas do modelo produzem saídas com a mesma dimensão d_{model} , para viabilizar essas conexões.

O decodificador (*decoder*) consiste em uma pilha de n camadas idênticas, cada uma com três subcamadas. A primeira subcamada é um mecanismo de atenção com máscara (*Masked Multi-Head Attention*) que impede determinadas posições de verificarem posições subsequentes. A segunda subcamada consiste em uma atenção multi-cabeças (*Multi-Head Attention*) que considera a saída da pilha de *encoder*, enquanto a terceira subcamada é rede neural totalmente conectada (*FeedForward*). Similar ao *encoder*, conexões residuais seguidas de camadas de normalização (*Layer Norm*) são aplicadas entre esses componentes.

O mecanismo de *Multi-Head Attention* (Eq. 3 e 4) aprimora a atenção padrão ao executar h funções de atenção, ou “cabeças”, em paralelo. Cada cabeça opera sobre projeções lineares das *queries*, *keys* e *values* para dimensões d_k , d_k e d_v , onde $d_k = d_v = (d_{model}/h)$. Isso permite ao modelo focar em diferentes subespaços de representação. Assim, os resultados das h funções são concatenados e novamente projetados para formar a saída final.

$$Attention_{MHead}(Q, K, V) = Concat(head_1, \dots, head_h)W^O, \quad (3)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V), \quad (4)$$

onde $W^O \in \mathbb{R}^{hd_v \times d_{model}}$, $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$ e $W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ são as matrizes de parâmetros para as projeções.

As redes *FeedForward* (Eq. 5), aplicadas individualmente a cada posição, consistem em duas transformações lineares com uma função de ativação ReLU (Eq. 6) entre elas. Este componente introduz não linearidade no modelo, o que amplia a capacidade de aprender relações mais complexas entre os elementos da sequência.

$$FeedForward(x) = ReLU(xW_1 + b_1)W_2 + b_2. \quad (5)$$

$$ReLU(x) = \max(0, x). \quad (6)$$

Por fim, outros componentes complementam a arquitetura. Como o modelo não possui recorrência ou convolução, as codificações posicionais são somadas aos *embeddings* de entrada para injetar informação sobre a ordem ou a posição dos elementos na sequência. As camadas de *embedding* convertem os dados de entrada e saída em vetores de dimensão d_{model} , e uma função *softmax* é usada na saída final do decodificador para gerar uma distribuição de probabilidades da saída.

3. Materiais e Métodos

A Figura 1 apresenta a metodologia discutida neste trabalho. As seções seguintes detalham cada uma das etapas analisadas. Além disso, é válido ressaltar que todo código foi desenvolvido em Python 3.12.3, utilizando as bibliotecas PyTorch e Scikit-Learn.

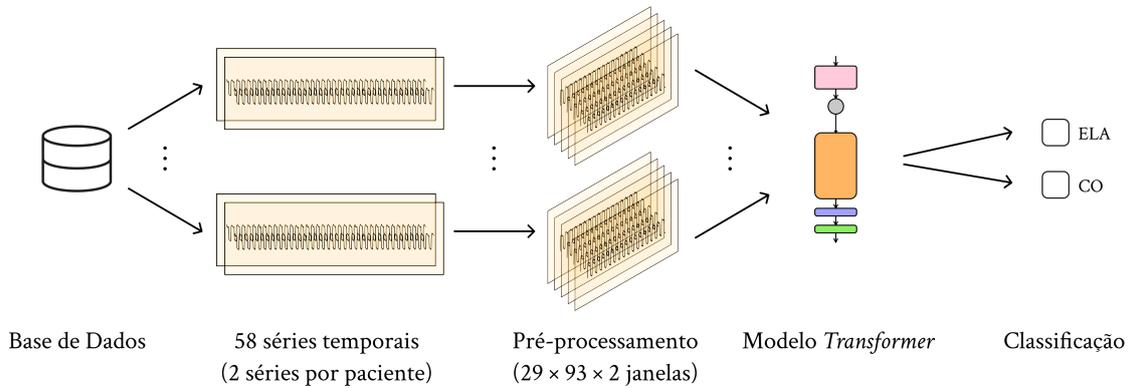


Figura 1. Fluxograma do método proposto.

3.1. Base de Dados

No presente trabalho, foi utilizada a *Gait in Neurodegenerative Disease Database* (GaitNDD), uma base de dados pública fornecida por Hausdorff et al. [1997, 2000] e disponível no sítio *PhysioNet*¹. A base contém dados coletados de até 5 minutos da marcha de 13 pessoas diagnosticadas com Esclerose Lateral Amiotrófica (10 homens e 3 mulheres) e 16 indivíduos saudáveis de controle (2 homens e 14 mulheres), totalizando 29 participantes.

Com sensores sensíveis de força [Hausdorff et al., 1995] acoplados aos pés (esquerdo e direito) dos voluntários, os sinais brutos de marcha foram coletados a uma taxa

¹<https://physionet.org/content/gaitnnd/1.0.0>

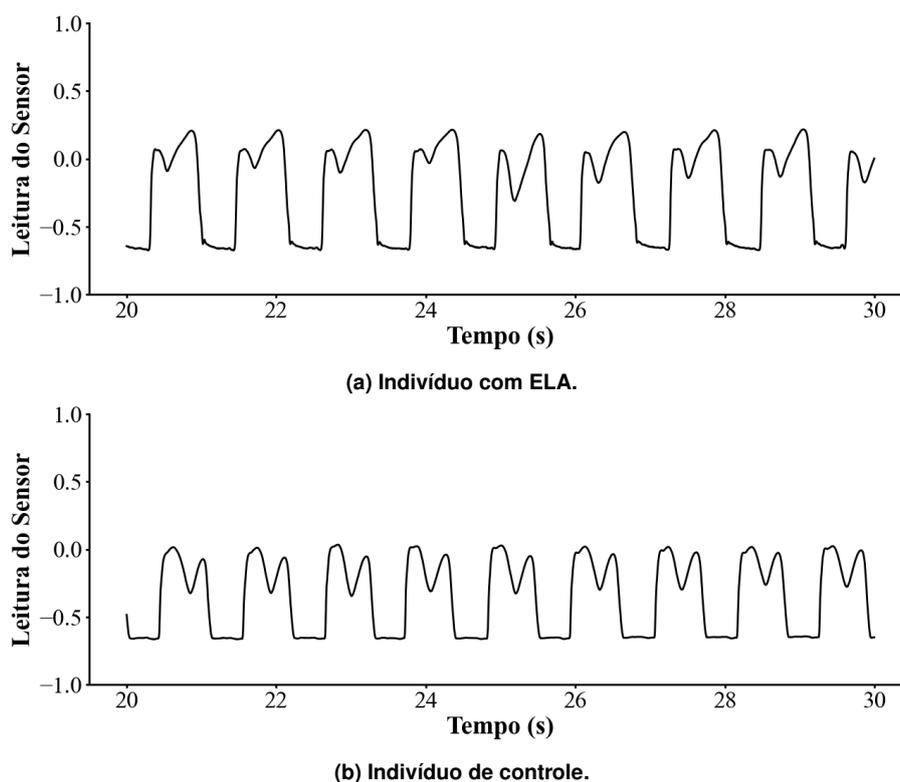


Figura 2. Amostra de dados de 10 segundos da marcha do pé esquerdo para indivíduos de classes distintas.

de amostragem de 300 Hz. Com isso, foi solicitado a cada participante que caminhasse durante 5 minutos, sem apoio e em ritmo autodeterminado, em um corredor de 77 m de comprimento. Isso resultou em duas séries temporais (uma para cada pé) de 90 mil pontos de dados ($300 \text{ Hz} \times 5 \times 60 \text{ segundos}$). Exemplos de 10 segundos desses dados brutos descritos são mostrados na Figura 2, na qual, em cada gráfico, o eixo- x representa o instante da caminhada em segundos e o eixo- y , a leitura dos sensores, a qual é proporcional à força que o indivíduo exerce sobre o solo.

3.2. Pré-processamento

Os autores da base de dados [Hausdorff et al., 1997, 2000] reportam efeitos de inicialização na coleta da marcha como ruídos. Por esse motivo, os dados coletados nos primeiros 20 segundos de cada série temporal foram descartados. Em seguida, os valores faltantes (*NaN values*) foram tratados, sendo substituídos pelo próximo valor válido na série e, ainda, qualquer valor NaN restante foi substituído pelo valor válido anterior.

Subsequentemente, uma técnica de janelamento que segmenta os sinais em janelas deslizantes foi utilizada. O tamanho da janela e o tamanho do passo (*step*) foram fixados em 3 segundos, de modo que não houvesse intersecção na segmentação. Essa técnica foi usada para maximizar a quantidade de dados provenientes de um mesmo participante. Por fim, os valores das janelas foram normalizados para valores de 0 a 3,5 [Hausdorff et al., 1995]. Assim, cada um dos 29 voluntários passa a ser representado por dois conjuntos de 93 janelas, cada uma com 900 pontos de dados. Portanto, os dados de entrada da rede *transformer* são representados por um vetor tridimensional ($29 \times 93, 900, 2$).

3.3. Classificação

Para realizar a tarefa de classificação, foi utilizada uma arquitetura *transformer* baseada apenas em codificador, mostrada na Figura 3.

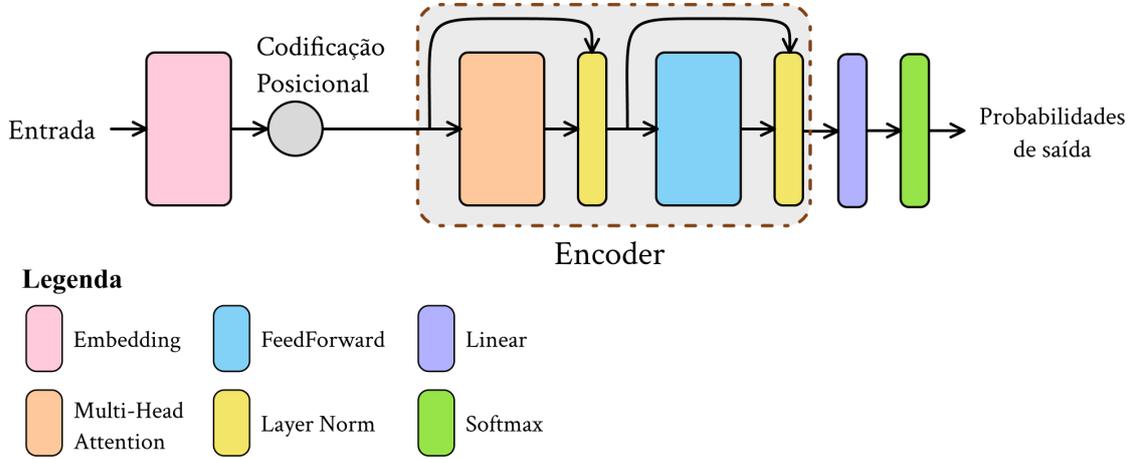


Figura 3. Arquitetura Transformer utilizada neste trabalho.

Na camada de *embedding*, o modelo projeta linearmente a dimensão de entrada 2 (representando os sinais de marcha obtidos dos dois pés) em um espaço de *embedding* de $d_{model} = 64$. Essa projeção é aplicada para capturar o significado dos elementos em um espaço de maior dimensão e para suportar todas as conexões residuais do *transformer*. Em seguida, o modelo aplica aos dados uma codificação posicional, que alterna entre funções seno e cosseno de diferentes frequências para fornecer informações posicionais exclusivas para cada elemento na sequência de entrada, detalhada pelas Eq. 7 e 8, onde pos representa a posição do elemento dentro da sequência de entrada e i representa o índice da dimensão dentro do vetor resultante da camada de *embedding* ($0 \leq i \leq d_{model} = 64$). Esse valores posicionais são incorporados aos *embeddings* de entrada para garantir que o *encoder* possa utilizá-los.

$$PE(pos, i) = \begin{cases} \sin(pos \times div_{term}(i)), & \text{se } i \text{ é par.} \\ \cos(pos \times div_{term}(i)), & \text{se } i \text{ é ímpar.} \end{cases} \quad (7)$$

$$div_{term}(i) = \exp\left(\frac{-i \times \ln 10,000}{d_{model}}\right). \quad (8)$$

Os dados incorporados e codificados são subsequentemente processados por uma pilha de $n = 2$ camadas do codificador. Dentro de cada uma dessas camadas, a subcamada de *Multi-Head Attention* opera com $h = 4$ cabeças de atenção, e a subcamada *feedforward* possui dimensionalidade de 2,048. O *dropout*, configurado para 0,1, é aplicado após o mecanismo de atenção e em cada etapa da *feedforward* para mitigar o *overfitting*.

Após o codificador, a representação da sequência é condensada a um único vetor por meio de uma operação de média (*mean pooling*) ao longo da dimensão temporal. Por fim, a representação reduzida passa por uma camada de normalização (*Layer Norm*) seguida por uma camada linear, resultando nos *logits* para a tarefa de classificação de duas classes. Esses *logits* passam por uma camada de *softmax* e tornam-se representações

das probabilidades de classificação. A fase de treinamento foi realizada com 50 épocas, utilizando entropia cruzada para a função de perda e Adam para a função do otimizador, com uma taxa de aprendizado de 0,001.

3.4. Avaliação

Os dados pré-processados foram divididos em conjuntos de treinamento (80%) e teste (20%). A fim de evitar o vazamento de dados (*data leakage*), a separação foi realizada por indivíduo, o que assegura que todas as janelas de dados gerados de um mesmo participante pertençam exclusivamente a um único conjunto.

O desempenho do modelo foi aferido por meio de duas métricas: a matriz de confusão, para uma análise visual das classificações; a acurácia, calculada como a razão entre o total de classificações corretas sobre o total de amostras avaliadas; a sensibilidade, que quantifica a proporção de positivos verdadeiros corretamente identificados; e a especificidade, que quantifica a proporção de negativos verdadeiros corretamente identificados.

4. Resultados

O modelo proposto alcançou uma acurácia final de 92,47%, sensibilidade de 91,76% e especificidade de 93,19% na tarefa de classificação binária para identificar os indivíduos com Esclerose Lateral Amiotrófica (ELA). Esses altos valores nas métricas, juntamente com a abordagem cautelosa de separação dos dados, destaca a eficácia do método baseado em *transformers* na identificação de Esclerose Lateral Amiotrófica a partir da marcha.

Tabela 1. Matriz de confusão que representa a acurácia de cada classe.

		Valores Preditos	
		CO	ELA
Valores Reais	CO	93,19%	6,81%
	ELA	8,24%	91,76%

Para uma análise mais detalhada dos resultados da classificação, uma matriz de confusão foi gerada. A Tabela 1 apresenta a matriz de confusão normalizada para cada classe. Essa matriz exibe uma forte diagonal principal, indicando que o modelo classifica corretamente a maioria dos indivíduos em suas respectivas categorias, com mínimas classificações incorretas.

5. Discussão

Os resultados obtidos neste trabalho, que alcançaram uma acurácia de 92,47%, são competitivos e comparáveis aos valores reportados na literatura para a mesma tarefa de classificação, conforme detalhado na Tabela 2. Embora a acurácia seja numericamente inferior a algumas abordagens, é válido destacar a principal vantagem metodológica do modelo proposto. O *Transformer* opera diretamente sobre os dados brutos da marcha, o que elimina a necessidade de uma etapa complexa e manual de engenharia e extração de características, um pré-requisito para métodos listados [Bilgin and Akin, 2018; Elden

et al., 2018; Gupta et al., 2019; Prabhu et al., 2020]. Esta abordagem não apenas simplifica todo o processo de análise, como também mitiga o risco de introduzir vieses no processo de seleção de características.

Tabela 2. Comparação entre as acurácias de métodos da literatura para o diagnóstico de Esclerose Lateral Amiotrófica em uma tarefa de classificação.

Trabalho	Classificador	Acurácia
Bilgin and Akin [2018]	MLP	96,55%
Elden et al. [2018]	SVM (RBF)	86,21%
Gupta et al. [2019]	Árvore de Decisão	96,20%
Prabhu et al. [2020]	SVM (RBF)	96,15%
Beyrami and Ghaderyan [2020]	NNLS Esparça (RBF)	100,00%
Este trabalho	Modelo Transformer	92,47%

Além disso, uma análise crítica dos trabalhos com as maiores acurácias, como o de [Beyrami and Ghaderyan, 2020] que reporta 100%, revela uma potencial limitação. O referido estudo não especifica se a divisão dos dados entre treinamento e teste foi realizada a nível de indivíduo, como feito neste trabalho. Em métodos que utilizam janelas de tempo para extrair *features*, a ausência de uma separação estrita por participante na validação pode ocasionar o vazamento de dados (*data leakage*). Isso já vem sendo reportado por estudos anteriores [Chagas et al., 2024, 2025], os quais destacam que a existência de informações temporais de um mesmo indivíduo simultaneamente nos conjuntos de treino e teste podem inflar artificialmente as métricas de desempenho.

Portanto, a solução baseada em *Transformer* apresentada neste trabalho surge como uma alternativa promissora, que oferece um desempenho robusto com uma metodologia mais simples e uma estratégia de validação que previne explicitamente o vazamento de dados.

6. Conclusão

Este trabalho propôs uma nova abordagem para a detecção de Esclerose Lateral Amiotrófica utilizando um modelo de arquitetura baseada em *transformer*. Para atingir esse objetivo, foram analisadas séries temporais obtidas dos pés esquerdo e direito de indivíduos diagnosticados com Esclerose Lateral Amiotrófica (ELA) e de indivíduos de controle (CO), durante uma caminhada de 5 minutos sem apoio. Os sinais obtidos de cada pé foram segmentados em janelas menores e processados por um modelo *transformer* para realizar uma tarefa de classificação binária (ELA vs CO).

O método proposto obteve uma acurácia de 92,47%, evidenciando desempenho comparativo em relação a literatura e trazendo as vantagens da rede *transformer*. Além disso, outra contribuição deste trabalho consiste no emprego de uma separação não enviesada de dados de treinamento e teste ao se utilizar a técnica de janelamento de dados,

frequentemente desconsiderada por outros trabalhos. Embora a generalização do método proposto seja limitada devido ao total de pessoas na base de dados, mesmo com a estratégia de janelamento, esses resultados indicam que o uso dessa arquitetura, em conjunto com a análise dos dados de marcha, pode favorecer o diagnóstico automático de ELA. Dessa forma, a abordagem tem potencial para viabilizar a identificação precoce da condição, permitindo o início antecipado do tratamento, o que pode contribuir para desacelerar a progressão da doença e promover uma melhor qualidade de vida ao paciente.

Agradecimentos

O presente trabalho foi realizado com apoio do Conselho Nacional de Desenvolvimento Científico (CNPq) e da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Código de Financiamento 001, e Edital N° 30/2022 – Programa Emergencial de Solidariedade Acadêmica (PDPG).

Referências

- Agarwal, K., Dheekollu, L., Dhama, G., Arora, A., Asthana, S., and Bhowmik, T. (2020). Deep learning based time series forecasting. In *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 859–864.
- Berman, T. and Bayati, A. (2018). What are Neurodegenerative Diseases and How Do They Affect the Brain? *Frontiers for Young Minds*, 6.
- Beyrami, S. M. G. and Ghaderyan, P. (2020). A robust, cost-effective and non-invasive computer-aided method for diagnosis three types of neurodegenerative diseases with gait signal analysis. *Measurement*, 156:107579.
- Bilgin, S. and Akin, Z. E. (2018). Gait pattern discrimination of als patients using classification methods. *Turkish Journal of Electrical Engineering and Computer Sciences*, 26(3):1367–1377.
- Bucci, G. D. F. F. B., Felix, J., Salvini, R., Nascimento, H., and Soares, F. (2025). Encoder-only transformer for detecting multiple neurodegenerative diseases from gait analysis. In *2025 IEEE 49th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 907–912.
- Chagas, A., Bucci, G., Félix, J., Fonseca, A., Nascimento, H., and Soares, F. (2024). Avaliando a sobreamostragem de dados temporais de marcha no diagnóstico automático de doenças neurodegenerativas. In *Anais do XXIV Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS)*, pages 567–578, Porto Alegre, RS, Brasil. SBC.
- Chagas, A., Bucci, G., Félix, J., Salvini, R., Nascimento, H., and Soares, F. (2025). Exploring biases in machine learning models for neurodegenerative diseases diagnosis through gait and voice analysis. In *Anais Estendidos do XXV Simpósio Brasileiro de Computação Aplicada à Saúde*, pages 1–6, Porto Alegre, RS, Brasil. SBC.
- Elden, R. H., Al-Atabany, W., and Ghoneim, V. F. (2018). Gait variability analysis in neurodegenerative diseases using nonlinear dynamical modelling. In *2018 9th Cairo International Biomedical Engineering Conference (CIBEC)*, pages 41–44.
- Erkkinen, M. G., Kim, M.-O., and Geschwind, M. D. (2018). Clinical neurology and epi-

- demology of the major neurodegenerative diseases. *Cold Spring Harbor Perspectives in Biology*, 10(4):a033118.
- Goldfarb, B. and Simon, S. (1984). Gait patterns in patients with amyotrophic lateral sclerosis. *Archives of physical medicine and rehabilitation*, 65(2):61—65.
- Gupta, K., Khajuria, A., Chatterjee, N., Joshi, P., and Joshi, D. (2019). Rule based classification of neurodegenerative diseases using data driven gait features. *Health and Technology*, 9(4):547–560.
- Hardiman, O., Van Den Berg, L. H., and Kiernan, M. C. (2011). Clinical diagnosis and management of amyotrophic lateral sclerosis. *Nature reviews neurology*, 7(11):639–649.
- Hausdorff, J. M., Ladin, Z., and Wei, J. Y. (1995). Footswitch system for measurement of the temporal parameters of gait. *Journal of Biomechanics*, 28(3):347–351.
- Hausdorff, J. M., Lertratanakul, A., Cudkowicz, M. E., Peterson, A. L., Kaliton, D., and Goldberger, A. L. (2000). Dynamic Markers of Altered Gait Rhythm in Amyotrophic Lateral Sclerosis. *Journal of Applied Physiology*, 88(6):2045–2053.
- Hausdorff, J. M., Mitchell, S. L., Firtion, R., Peng, C.-K., Cudkowicz, M. E., Wei, J. Y., and Goldberger, A. L. (1997). Altered fractal dynamics of gait: reduced stride-interval correlations with aging and Huntington’s disease. *Journal of Applied Physiology*, 82(1):262–269.
- Heemels, M.-T. (2016). Neurodegenerative diseases. *Nature*, 539(7628):179–180.
- Masrori, P. and Van Damme, P. (2020). Amyotrophic lateral sclerosis: a clinical review. *European journal of neurology*, 27(10):1918–1929.
- Mayeux, R. (2003). Epidemiology of neurodegeneration. *Annual Review of Neuroscience*, 26(1):81–104.
- Naimi, S., Bouachir, W., and Bilodeau, G.-A. (2023). 1d-convolutional transformer for parkinson disease diagnosis from gait. *Neural Computing and Applications*, 36(4):1947–1957.
- Nguyen, D. M. D., Miah, M., Bilodeau, G.-A., and Bouachir, W. (2022). Transformers for 1d signals in parkinson’s disease detection from gait. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 5089–5095.
- Prabhu, P., Karunakar, A., Anitha, H., and Pradhan, N. (2020). Classification of gait signals into different neurodegenerative diseases using statistical analysis and recurrence quantification analysis. *Pattern Recognition Letters*, 139:10–16.
- Sun, H.-J. and Zhang, Z.-G. (2022). Transformer-based severity detection of parkinson’s symptoms from gait. In *2022 15th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 1–5.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.