

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

CARLOS GABRIEL DE FREITAS
Orientador: Pedro Henrique Lopes Silva

**APLICAÇÃO DE *DEEP LEARNING* PARA BIOMETRIA BASEADA EM
EEG NO MODO DE IDENTIFICAÇÃO**

Ouro Preto, MG
2023

UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

CARLOS GABRIEL DE FREITAS

**APLICAÇÃO DE *DEEP LEARNING* PARA BIOMETRIA BASEADA EM EEG NO
MODO DE IDENTIFICAÇÃO**

Monografia II apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Pedro Henrique Lopes Silva

Ouro Preto, MG
2023

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

F866a Freitas, Carlos Gabriel De.
Aplicação de Deep Learning para Biometria Baseada em EEG no Modo de Identificação. [manuscrito] / Carlos Gabriel De Freitas. - 2023.
42 f.: il.: color., tab..

Orientador: Prof. Dr. Pedro Henrique Lopes Silva.
Monografia (Bacharelado). Universidade Federal de Ouro Preto.
Instituto de Ciências Exatas e Biológicas. Graduação em Ciência da Computação .

1. Biometria. 2. Identificação. 3. Aprendizagem em Profundidade. I. Silva, Pedro Henrique Lopes. II. Universidade Federal de Ouro Preto. III. Título.

CDU 004

Bibliotecário(a) Responsável: Luciana De Oliveira - SIAPE: 1.937.800



FOLHA DE APROVAÇÃO

Carlos Gabriel de Freitas

Aplicação de Deep Learning para Biometria Baseada em EEG no Modo de Identificação

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 22 de Março de 2023.

Membros da banca

Pedro Henrique Lopes Silva (Orientador) - Doutor - Universidade Federal de Ouro Preto
Eduardo José da Silva Luz (Examinador) - Doutor - Universidade Federal de Ouro Preto
Rafael Alves Bonfim de Queiroz (Examinador) - Doutor - Universidade Federal de Ouro Preto

Pedro Henrique Lopes Silva, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 22/03/2023.



Documento assinado eletronicamente por **Pedro Henrique Lopes Silva, PROFESSOR DE MAGISTERIO SUPERIOR**, em 22/03/2023, às 16:39, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **0493044** e o código CRC **6280CC78**.

Dedico este trabalho a Deus, meus pais, meu irmão e meus amigos.

Agradecimentos

A Deus, pela vida que me foi dada e pela minha saúde.

Aos meus pais e meu irmão, pelo apoio que me foi dado durante a graduação e na decisão de realizá-la.

Aos meus amigos, por me acompanharem e auxiliarem nos obstáculos desta caminhada.

Ao meu orientador, pela disposição em realizar um bom trabalho.

“Remember to look up at the stars and not down at your feet. Try to make sense of what you see and wonder about what makes the universe exist. Be curious. And however difficult life may seem, there is always something you can do and succeed at. It matters that you don’t just give up.”

(HAWKING, 2010)

Resumo

Com o avanço da biometria e a necessidade de sistemas de segurança mais robustos, outros tipos de características humanas além das mais utilizadas foram levadas em consideração no desenvolvimento de sistemas biométricos. Uma destas características é o eletroencefalograma (sinais cerebrais). Este trabalho então avalia diferentes arquiteturas de redes neurais no cenário de identificação biométrica: *Convolutional Neural Network* (CNN), utilização de blocos contíguos de *Long Short-Term Memory* (LSTM) e *Gated Recurrent Unit* (GRU), e fusão entre CNN e LSTM. Um resultado de 99,81% de acurácia foi atingido ao utilizar *padding* causal nas camadas de convolução do modelo que funde as arquiteturas CNN e LSTM, um tamanho de janela de 12 segundos e ao compor os dados de treinamento com sete tarefas diferentes, realizadas pelos 109 indivíduos presentes na base de dados EEG *Motor Movement/Imagery Dataset* da PhysioNet.

Palavras-chave: Aprendizagem em profundidade. Biometria. Identificação. Eletroencefalograma. CNN-LSTM.

Abstract

With the advancement of biometrics and the necessity of more robust security systems, other types of human characteristics other than the most used were taken in consideration on the development of biometric systems. One of these characteristics is the electroencephalogram (brain signals). This work evaluates different neural network architectures on the biometric identification scenery: Convolutional Neural Network (CNN), usage of contiguous Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) blocks, and a fusion between CNN and LSTM. A result of 99,81% of accuracy were attained when using causal padding on the convolution layers of the model that fuses CNN and LSTM architectures, a 12-second window size and composing the training data with seven different tasks, done by the 109 individuals present in the EEG Motor Movement/Imagery Dataset database from PhysioNet.

Keywords: Deep Learning. Biometrics. Identification. Electroencephalogram. CNN-LSTM.

Lista de Ilustrações

Figura 2.1 – Principais modos de operação de um sistema biométrico.	6
Figura 2.2 – Visualização de um neurônio.	7
Figura 2.3 – Exemplificações do impacto causado por diferentes <i>learning rates</i>	8
Figura 2.4 – Estrutura de uma CNN.	9
Figura 2.5 – Exemplificação do uso de <i>padding</i> causal.	10
Figura 2.6 – Estrutura de uma célula LSTM e de uma célula GRU.	11
Figura 2.7 – Posicionamento dos eletrodos no couro cabeludo.	12
Figura 2.8 – Diferenciação entre sinais de indivíduos diferentes em estado de repouso. . .	13
Figura 2.9 – Diferenciação entre sinais de um mesmo indivíduo executando tarefas diferentes.	13
Figura 2.10–Visualização de um EER.	16
Figura 3.1 – Metodologia desenvolvida no trabalho proposto.	21
Figura 3.2 – Diferenciação entre um sinal não filtrado e um filtrado.	22
Figura 3.3 – <i>Data augmentation</i> com janela deslizante.	23
Figura 3.4 – Modo de composição das tarefas de treino, validação e teste utilizado no trabalho proposto.	24
Figura 4.1 – Arquitetura final adotada.	31

Lista de Tabelas

Tabela 3.1 – Arquitetura da CNN utilizada inicialmente.	25
Tabela 3.2 – Arquitetura que utiliza LSTM/GRU utilizada inicialmente.	25
Tabela 4.1 – Filtro e normalização no pré-processamento dos dados.	28
Tabela 4.2 – Tamanhos de <i>offset</i> para o processo de <i>data augmentation</i>	29
Tabela 4.3 – Valores de épocas em que o modelo é treinado.	29
Tabela 4.4 – Utilizando blocos de LSTM e GRU contíguos.	29
Tabela 4.5 – Utilizando blocos de LSTM e GRU bidirecionais, unidos à arquitetura da CNN apresentada anteriormente.	30
Tabela 4.6 – Variando o número de unidades por bloco de LSTM e GRU bidirecional. . .	30
Tabela 4.7 – Variando o número de blocos de LSTM.	31
Tabela 4.8 – Resultado para o protocolo P1.	32
Tabela 4.9 – Resultado para o protocolo P2.	33
Tabela 4.10–Resultado para o protocolo P3.	34
Tabela 4.11–Utilização de <i>padding</i> causal nas camadas de convolução.	34
Tabela 4.12–Comparação entre trabalhos com foco modo de verificação.	34
Tabela 4.13–Comparação entre trabalhos com foco modo de identificação.	35

Lista de Abreviaturas e Siglas

CNN	<i>Convolutional Neural Network</i>
EEG	Eletroencefalograma
FIR	<i>Finite Impulse Response</i>
GRU	<i>Gated Recurrent Unit</i>
IIR	<i>Infinite Impulse Response</i>
LSTM	<i>Long Short-Term Memory</i>
ReLU	<i>Rectified Linear Unit</i>
REO	<i>Rest Eyes Open</i>
REC	<i>Rest Eyes Closed</i>
RNN	<i>Recurrent Neural Network</i>
SGD	<i>Stochastic Gradient Descent</i>
tanh	Tangente Hiperbólica

Lista de Símbolos

Σ	Letra grega maiúscula Sigma que representa um somatório.
e	Letra latina minúscula E que representa o número de Euler.
S	Letra latina maiúscula S que representa a função de ativação <i>softmax</i> .
$\#$	Cerquilha que representa a quantidade de algo.

Sumário

1	Introdução	1
1.1	Justificativa	2
1.2	Objetivos	2
1.3	Organização do Trabalho	3
2	Revisão Bibliográfica	4
2.1	Fundamentação Teórica	4
2.1.1	Biometria	4
2.1.2	Aprendizado em Profundidade (<i>Deep Learning</i>)	6
2.1.3	Base de Dados <i>PhysioNet</i>	11
2.1.4	Filtragem de Dados	13
2.1.5	Métricas Utilizadas	15
2.2	Trabalhos Relacionados	16
3	Desenvolvimento	21
3.1	Pré-Processamento	22
3.2	<i>Data Augmentation</i>	23
3.3	Composição dos Dados de Validação	24
3.4	Arquiteturas Utilizadas	24
3.5	Avaliação da Rede	26
4	Experimentos e Resultados	27
4.1	Filtragem e Normalização	27
4.2	Avaliação dos Parâmetros	28
4.3	Utilização de Blocos LSTM e GRU	29
4.4	Protocolos P1, P2 e P3	31
4.5	<i>Padding Causal</i>	33
4.6	Comparação com os trabalhos relacionados	34
4.7	Discussões	35
4.7.1	Performance de uma arquitetura CNN e LSTM/GRU e LSTM+CNN	35
4.7.2	Performance por região cerebral	35
4.7.3	Composição das tarefas de treino	36
4.7.4	Utilização de <i>padding</i> causal	36
5	Considerações Finais	37
5.1	Conclusão	37
5.2	Trabalhos Futuros	37
5.3	Publicações Realizadas	38

Referências 39

1 Introdução

O desenvolvimento da biometria trouxe consigo o avanço tecnológico dos sistemas de identificação e de segurança. O acesso a caixas e urnas eletrônicas, por exemplo, passaram do uso de senhas, cartões e documentos impressos para o uso de digitais, as quais oferecem mais comodidade às pessoas ao eliminar o risco de esquecerem os elementos usados para sua identificação (LUMINI; NANNI, 2017). A utilização da biometria também continua permitindo o uso do serviço pelas pessoas em casos de roubo, e possuem menos chance de serem falsificadas (OBAIDAT et al., 2019).

Retina, voz e rosto, dentre outras características humanas que diferem de indivíduo para indivíduo, assim como combinações entre estas características, também se tornaram opções válidas para serem utilizados em sistemas biométricos. Assim, os sistemas se tornam mais robustos e menos suscetíveis a falsos-positivos (sistema permite o acesso a um indivíduo que se passa por outro) (SARIN et al., 2021).

Sistemas biométricos podem ser utilizados para combater fraude (BOBKOWSKA; NAGATY; PRZYBORSKI, 2019) e aumentar o nível de segurança de serviços de saúde, transporte e sociais, dentre outros tipos, utilizados diariamente por um grande número de pessoas (OBAIDAT et al., 2019). Pelos motivos expostos, tais sistemas vêm ganhando cada vez mais atenção dos pesquisadores.

Uma das características humanas que vem sendo exploradas para construção de sistemas biométricos é o eletroencefalograma (EEG) (CARRIÓN-OJEDA; FONSECA-DELGADO; PINEDA, 2021; SCHONS et al., 2017), que consiste no registro dos sinais elétricos presentes na atividade cerebral através do posicionamento de eletrodos no couro cabeludo. Os sinais de um EEG variam de acordo com o modo como um indivíduo executa uma tarefa, seja ela motora ou imaginária, bem como o estado emocional em que ele se encontra (LEE; HSIEH, 2014).

Ademais, já que é necessário uma série de procedimentos e equipamentos específicos para capturar sinais de EEG, sistemas biométricos baseados neles se tornam menos suscetíveis a fraudes se comparados com outros sistemas que utilizam outra característica humana (MAIORANA, 2020).

A fim de extrair as características que diferenciam um indivíduo do outro, e que “residem” dentro de um sinal EEG, podemos utilizar diferentes técnicas, como calcular a distância de Hamming entre dois sinais codificados (DAMAŠEVIČIUS et al., 2018), similaridade por cosseno (DAS; MAIORANA; CAMPISI, 2016), ou um tipo de rede neural explorado na área de *Deep Learning* chamado *Convolutional Neural Network* (CNN) (SCHONS et al., 2017), que é focado em realizar operações de convolução.

Outra arquitetura, presente em redes neurais, envolve a utilização de camadas de *Long Short-Term Memory* (LSTM). Este tipo de arquitetura apresenta bons resultados em experimentos que possuem dados sequenciais (SHERSTINSKY, 2020), e além de ser utilizada em tradutores de texto (WU; QIN, 2022) e modelagem linguística (LIN; TEGMARK, 2017), também podem ser utilizada na construção de sistemas biométricos (SUN; LO; LO, 2019).

Este trabalho então propõe avaliar, para o contexto de identificação biométrica, uma CNN baseada no trabalho de (SCHONS et al., 2017), a utilização de blocos de LSTM e *Gated Recurrent Unit* (GRU) e o fusionamento destes elementos. Serão utilizados dados dos 109 indivíduos presentes na base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004), pertencente à *PhysioNet* (GOLDBERGER et al., 2000).

1.1 Justificativa

Como pode ser visto na introdução, a biometria é presente em muitos aspectos da vida cotidiana, e se demonstra como uma ferramenta capaz de aumentar o nível de segurança de sistemas de saúde, transporte e sociais, dentre outros tipos, através do uso de diversas características humanas. Com o advento e popularização de sistemas e aplicativos modernos, se tem o risco de pessoas mal-intencionadas tentarem fraudar tais sistemas, e portanto, se faz cada vez mais necessário o desenvolvimento de sistemas biométricos mais robustos.

Além de poder ser utilizado para biometria, o EEG também pode ser utilizado para detectar variações emocionais de uma pessoa. Por sua vez, tal detecção poderia ser aplicada para verificar inconsistências na fala de uma pessoa em investigações, ou em checagens psicológicas (ISMAIL et al., 2016).

Ademais, o presente trabalho foi parte de um projeto de iniciação científica. Como forma de dar continuidade, testes adicionais foram realizados, com o objetivo de melhorar os resultados obtidos pela rede neural.

1.2 Objetivos

O objetivo principal deste trabalho é utilizar uma *Convolutional Neural Network* (CNN) e uma arquitetura baseada em blocos de *Long Short-Term Memory* (LSTM) ou *Gated Recurrent Unit* (GRU) para construir um sistema biométrico baseado em eletroencefalograma, com foco no modo de identificação. Como objetivos específicos, pretendem-se:

- Analisar os resultados obtidos por uma CNN, e redes compostas por blocos de LSTM ou GRU contíguos.
- Explorar o fusionamento de blocos LSTM e GRU ao modelo de CNN inicialmente proposto.

- Avaliar como a composição das tarefas utilizadas durante as etapas de treino e teste da rede neural afetam a acurácia apresentada por ela, através dos protocolos P1, P2 e P3 propostos por Yang, Deravi e Hoque (2018).
- Avaliar o impacto do uso de *padding* causal nas camadas de convolução presentes no modelo neural.

1.3 Organização do Trabalho

O restante desta monografia é organizada da seguinte maneira: O Capítulo 2 (**Revisão Bibliográfica**) apresenta a contextualização da pesquisa realizada, explicando conceitos abordados e destacando outros trabalhos relacionados com o proposto; o Capítulo 3 (**Desenvolvimento**) apresenta a metodologia adotada; o Capítulo 4 (**Resultados**) demonstra os resultados obtidos nos diferentes experimentos realizados, e discussões acerca deles; e por fim, o Capítulo 5 (**Considerações Finais**) apresenta as conclusões tomadas a partir do trabalho apresentado, bem como perspectiva de trabalhos futuros.

2 Revisão Bibliográfica

Este capítulo apresenta a contextualização da pesquisa realizada e é dividido em duas seções: Seção 2.1 (**Fundamentação Teórica**) e Seção 2.2 (**Trabalhos Relacionados**).

Na Seção 2.1, são apresentados conceitos abordados nessa monografia que são de fundamental importância para a compreensão da mesma.

Na Seção 2.2, são apresentados trabalhos da literatura relacionados ao tema proposto. Os critérios para seleção dos trabalhos relacionados foram os que utilizavam CNN na construção de um sistema biométrico baseado em EEG, sendo as únicas exceções o trabalho de [Yang, Deravi e Hoque \(2018\)](#), já que este contém os protocolos que serão avaliados pelo trabalho proposto, e o trabalho de [Oord et al. \(2016\)](#), cuja utilização de convoluções causais também foi explorada no presente trabalho.

2.1 Fundamentação Teórica

2.1.1 Biometria

Diferentes aspectos físicos, comportamentais e químicos humanos podem ser utilizados para a criação de um sistema biométrico ([LUMINI; NANNI, 2017](#)). No entanto, de acordo com [Jain, Bolle e Pankanti \(2006\)](#), alguns fatores devem ser levados em consideração ao utilizar alguma característica humana para autenticação biométrica. São eles:

- **Universalidade:** cada indivíduo que usar o sistema deve possuir a característica escolhida. Desta forma, a autenticação de indivíduos por digitais não seria efetiva caso uma parte significativa deles não possuíssem as mãos ou pés;
- **Distinguilidade:** a característica deve diferir de indivíduo para indivíduo. Fios de cabelo, por exemplo, não diferem o suficiente para que sejam utilizados em sistemas biométricos;
- **Invariância ao longo do tempo:** o quanto a característica varia, ou não, com o tempo. Unhas, por exemplo, crescem e são cortadas em um ritmo muito alto, e seriam péssimas candidatas para a característica de um sistema biométrico neste fator;
- **Mensurabilidade:** o quão fácil é adquirir dados daquela característica e extrair seus pontos mais relevantes. Por exemplo, o equipamento necessário para capturar e processar eletrocardiogramas é mais caro do que os necessários para digitais;
- **Performance:** o quão veloz, preciso e robusto é o sistema biométrico. É possível que um sistema biométrico sacrifique sua velocidade de identificação de um indivíduo para que

esta seja mais precisa, o que não necessariamente implica que ele será pior que um outro sistema biométrico mais veloz;

- **Aceitabilidade:** o quão dispostos os indivíduos estão em aceitar a tecnologia e ter os dados das suas características armazenadas e acessadas. Se alguma das etapas descritas causar dor ou desconforto às pessoas, bem como infringir sua privacidade, é mais provável que elas não a aceitem;
- **Evasão:** o quão fácil a característica pode ser imitada ou fraudada. O processo de falsificação de digitais, por exemplo, é mais fácil que o de retinas.

Ainda, um sistema biométrico pode operar em três principais modos distintos (LUMINI; NANNI, 2017), os quais podem ser visualizados na Figura 2.1:

- **Registro:** insere um novo indivíduo no banco de dados do sistema. Seus dados são capturados por sensores e armazenados após técnicas de extração de características serem realizadas nestes dados.
- **Identificação:** este modo tenta responder à pergunta “Quem é esse indivíduo?”. Os dados de entrada se referem à um indivíduo desconhecido pela rede, e são comparados com todas as identidades registradas no sistema em um esquema um-para-todos.
- **Verificação:** este modo tenta responder à pergunta “Esse indivíduo é quem proclama ser?”. Os dados de entrada se referem à um indivíduo já registrado na rede, e são comparados com a sua identidade presente no sistema em um esquema um-para-um.

Apesar do alto custo do equipamento e mão de obra envolvidos na aquisição e processamento dos sinais de EEG, bem como na construção de um sistema biométrico baseado nestes sinais, esta característica apresenta uma alta robustez e altos resultados em ambos os modos de identificação e verificação, onde alguns destes últimos serão apresentados na Seção 2.2.

A utilização de EEG em sistemas biométricos também ocasiona na inclusão de pessoas com certos tipos de deficiência. Indivíduos com algum tipo de deficiência nos braços ou nos olhos, ou mesmo com a ausência destes, poderiam se beneficiar deste tipo de biometria sem necessitar de qualquer alteração no sistema.

A captura dos sinais de EEG, efetuada por eletrodos, também não causa dor aos usuários do sistema, e unido aos motivos citados anteriormente, sistemas biométricos baseados nesta característica podem possuir altas performance, universalidade e aceitabilidade, além de uma baixa evasão.

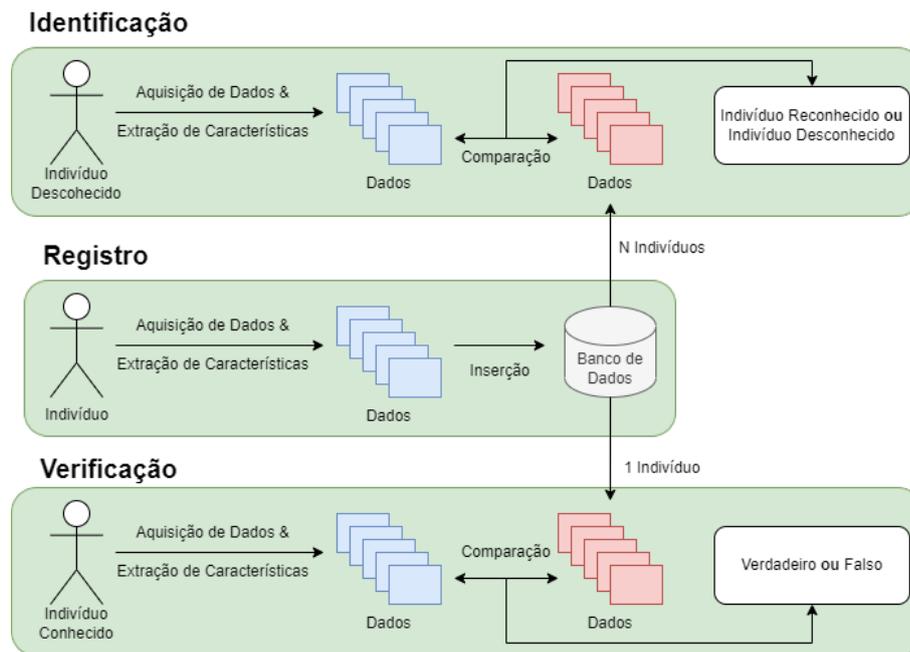


Figura 2.1 – Principais modos de operação de um sistema biométrico.
Fonte: Elaborado pelo autor.

2.1.2 Aprendizado em Profundidade (*Deep Learning*)

O aprendizado em profundidade, popularmente conhecido como *Deep Learning*, é um campo do aprendizado de máquina (*Machine Learning*), que incorpora várias camadas sucessivas de processamento para criar uma rede neural que irá aprender alguma tarefa designada a ela (LECUN; BENGIO; HINTON, 2015). Tal aprendizado é efetuado ao minimizar uma função de perda, a qual define a diferença entre a saída real e a saída gerada por um modelo neural referentes a uma mesma amostra de entrada.

Os dados de entrada de cada camada são somados e multiplicados por diferentes pesos, a fim de tornar mais relevantes certas características julgadas pela rede neural como importantes para definir a qual classe (no caso de um sistema biométrico, a qual indivíduo) aqueles dados pertencem, em um problema de classificação.

O componente básico de uma rede neural, o neurônio, pode ser visualizado na Figura 2.2. Na imagem, x_1 e x_2 seriam exemplos de dados de entrada do neurônio, os quais seriam respectivamente multiplicados pelos pesos w_1 e w_2 . Estes resultados então seriam somados com um valor de viés, também chamado de *bias*, representado por b , e uma função de ativação f seria então aplicada, resultando em um valor y como saída do neurônio.

Em aplicações reais, tanto os valores de entrada x quanto os pesos w de um neurônio são vetores N -dimensionais. Uma rede neural simples será tipicamente formada de várias camadas, onde cada uma destas camadas são formadas por vários neurônios, e as saídas dos neurônios de

¹ Imagem disponível em: <<https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks>>. Acesso em 13 de agosto de 2022.

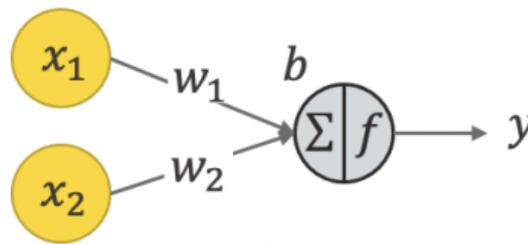


Figura 2.2 – Visualização de um neurônio.
Fonte: KNIME. ¹

uma camada serão entradas dos neurônios da próxima camada.

Os conjuntos de pesos de uma rede neural são ajustados a cada época, termo dado a uma passada em todos os dados de treinamento pela rede neural. Deste modo, se na atual fase do treinamento de um modelo neural, todos os dados de treinamento foram analisados e utilizados dez vezes, ele está na décima época. De forma a reduzir a quantidade de amostras que devem ser carregadas e analisadas pelo modelo por vez, pode-se dividir os dados de treinamento em grupos com um determinado número de amostras, chamados de *mini-batches* (ELIAS et al., 2020). Tal técnica foi utilizada no presente trabalho.

Para que o conjunto de pesos se ajuste de forma a se tornar um conjunto próximo da otimalidade para o problema abordado, ou seja, que apresente bons resultados com base nas métricas utilizadas, é necessário fazer uso de um algoritmo otimizador. O *Stochastic Gradient Descent* (ou SGD) é o algoritmo otimizador utilizado no presente no trabalho, que atualiza os pesos com base em uma quantidade de amostras de treino igual ao dos *mini-batches* por vez, em vez de todas elas, o que pode diminuir o custo computacional em problemas que possuam muitos exemplos de treino ou características a serem extraídas (KETKAR, 2017).

A etapa de atualização dos pesos é chamada de *backpropagation*, e é realizada no sentido contrário das camadas, ou seja, da última camada até a primeira. Nela, são calculadas as derivadas parciais de cada um dos pesos da rede neural em relação a função de perda, as quais irão indicar a taxa de variação desta função pelos pesos em questão. A partir de então, será possível construir o gradiente da função de perda, o qual indicará a direção para um ponto mínimo desta função (ROJAS, 1996). Tal etapa de atualização também faz uso de um parâmetro denominado *learning rate*, o qual irá dimensionar o tamanho do “passo” neste caminho indicado pelo gradiente.

Também pode ser definido um coeficiente de *momentum*, que tem por objetivo aumentar gradualmente o *learning rate* caso o algoritmo otimizador esteja caminhando na mesma direção, na tentativa de escapar de mínimos locais e encontrar o mínimo global da função de perda (SUTSKEVER et al., 2013).

Ao fazer uma analogia, podemos considerar o conjunto de pesos atual de um modelo neural como uma bola, a qual irá rolar pelo espaço de possíveis conjuntos de valores a qual este

pode assumir, representado por diferentes pontos de um plano. O *learning rate* seria a velocidade desta bola, determinando o quão rápido ou devagar este conjunto de pesos assume outros valores, e o *momentum* seria a aceleração desta bola, controlando o *learning rate*. A Figura 2.3 exemplifica o impacto causado ao escolher valores muito altos ou baixos para o *learning rate*.

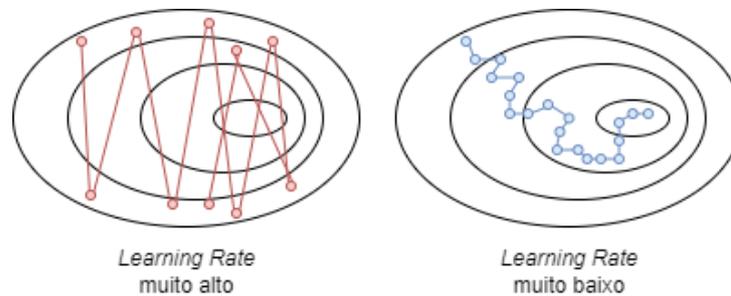


Figura 2.3 – Exemplificações do impacto causado por diferentes *learning rates*: à esquerda, um *learning rate* muito alto implica em saltos muito longos, e o conjunto de pesos pode não chegar à um mínimo local ou global; à direita, um *learning rate* muito baixo implica em saltos muito curtos, e uma grande quantidade de tempo pode se passar até que o conjunto de pesos chegue à um mínimo local ou global.

Fonte: Elaborado pelo autor.

A CNN é um tipo de rede neural que utiliza a operação de convolução de matrizes para realizar a extração de características (ALBAWI; MOHAMMED; AL-ZAWI, 2017), ou seja, reduzir e representar aspectos importantes de dados, como imagens, ou no caso do presente trabalho, eletroencefalogramas, ao utilizar um conjunto de valores numéricos. As camadas da rede neural responsáveis por realizar essa operação são chamadas de camadas de convolução.

As CNNs também contam com camadas de:

- *pooling*, que “agregam a informação presente em pequenas regiões dos seus dados de entrada” (SUN et al., 2017), por meio de operações como máximo ou média, entre os números presentes em cada região;
- ativação, que adicionam um comportamento não linear para os cálculos da rede;
- totalmente conectadas (também chamadas de densas), onde cada neurônio de duas camadas distintas são conectados;
- *dropout*, que podem desativar neurônios em uma camada aleatoriamente, fazendo assim com que os neurônios se tornem menos dependentes das informações processadas pelos outros (SRIVASTAVA et al., 2014);
- *batch normalization*, que aplicam um processo de normalização nos dados que recebem e o enviam para a próxima camada.

A função de ativação utilizada nas camadas de convolução e em algumas das camadas densas presentes neste trabalho é chamada de *Rectified Linear Unit*, ou ReLU. Esta função não apresenta o mesmo problema que outras funções de ativação, como a tangente hiperbólica (que será explicada posteriormente), chamado de *vanishing gradient*, onde para valores altos de entrada há um encolhimento dos valores presentes no gradiente, e assim, haverá pouca diferença na atualização dos pesos proporcionada pelo algoritmo otimizador (IDE; KURITA, 2017).

A ReLU é expressa por $ReLU(x) = \max(0, x)$, onde x é um dado de entrada de uma camada e \max é uma função que retorna o valor máximo dentre eles. A sua derivada é expressa pela equação:

$$ReLU'(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{caso contrário} \end{cases}, \quad (2.1)$$

onde x será um dado de entrada para a camada na etapa de *backpropagation* da rede neural.

A Figura 2.4 exemplifica a estrutura típica de uma CNN, contendo camadas de convolução, *pooling*, totalmente conectadas e funções de ativação.

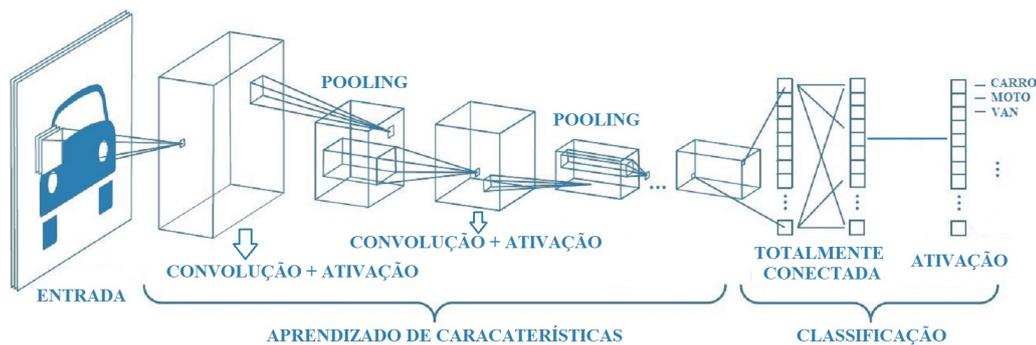


Figura 2.4 – Estrutura de uma CNN.

Fonte: Adaptado pelo autor.²

A função utilizada para “transformar” as saídas da última camada densa em probabilidades para cada uma das classes de uma rede neural é a *softmax*, cuja fórmula pode ser definida pela equação:

$$S(z)_i = \frac{e^{(z)_i}}{\sum_{j=1}^K e^{(z)_j}}, \quad (2.2)$$

onde z é um vetor de números, $(z)_i$ é o i -ésimo valor deste vetor, e representa o número de Euler com um valor aproximado a 2,71828, e K é o tamanho do vetor z .

A utilização de *padding* causal em camadas de convolução, também presente neste trabalho, consiste em colocar zeros à frente de um vetor de entrada, alterando as saídas produzidas por estas camadas de forma que o N -ésimo valor da saída tenha uma maior dependência do

² Modificação da imagem disponível em: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>. Acesso em 30 de setembro de 2021.

N-ésimo valor da entrada e dos valores que se encontram antes dele, criando assim uma relação de causalidade. A Figura 2.5 exemplifica a utilização de *padding* causal.

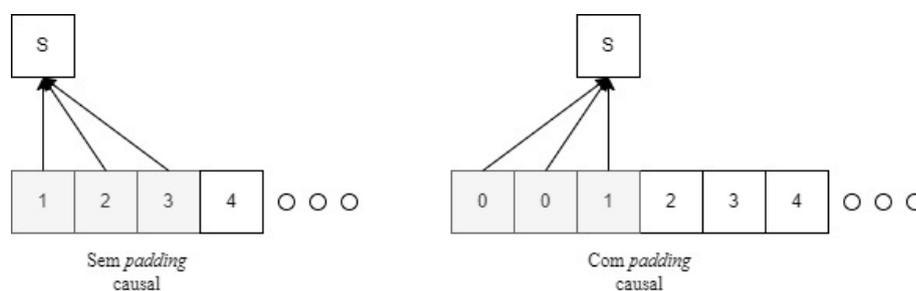


Figura 2.5 – Exemplificação do uso de *padding* causal. Suponha que, para a produção do primeiro dado de saída S, a rede utilize os três primeiros dados em um vetor de entrada. Sem o *padding* causal, a rede utilizaria os dados nomeados como 1, 2 e 3 para produzir S, e com o *padding* causal, zeros seriam colocados à frente do vetor de forma que apenas estes zeros e o dado nomeado como 1 fosse utilizado para tal.

Fonte: Elaborado pelo autor.

Já *Recurrent Neural Network* (RNN) é um outro tipo de rede neural que utiliza “células recorrentes cujos estados são afetados por estados prévios e pela entrada atual através de conexões de retroalimentação” (YU et al., 2019). Dois tipos principais destas células, também chamadas de blocos ou unidades, são a LSTM e a GRU.

Células LSTM possuem um terminal de entrada, que “decide qual informação pode ser armazenada em seu estado”, um terminal de esquecimento, que “decide se a informação armazenada continuará armazenada ou será jogada fora”, e um terminal de saída, que “decide qual informação sairá da célula baseado em seu estado” (YU et al., 2019). Já a célula GRU, “integra os terminais de entrada e esquecimento de uma LSTM em um único terminal de atualização” e possui um menor poder computacional do que uma célula LSTM devido ao seu menor número de terminais (YU et al., 2019). As estruturas de ambas as células, LSTM e GRU, podem ser visualizadas na Figura 2.6:

Segundo Yu et al. (2019), “Células LSTM e GRU convencionais só podem fazer uso do contexto anterior ao do que se encontram”, ou seja, se tivermos apenas três células LSTM ou GRU em uma rede neural, o processamento da informação na 2ª célula será baseado apenas na 1ª célula, que se encontra antes dela. Para solucionar este problema, foi desenvolvida a versão bidirecional das células citadas, que também conseguem fazer uso do contexto posterior ao do que se encontram (YU et al., 2019).

Esta característica das células bidirecionais demonstram uma vantagem em relação às células unidirecionais em contextos onde informações apresentadas posteriormente as que estão sendo analisadas têm uma alta contribuição para o resultado, como na classificação de fonemas (GRAVES; FERNÁNDEZ; SCHMIDHUBER, 2005).

³ Modificação da imagem disponível em: <<https://github.com/roomylee/rnn-text-classification-tf>>. Acesso em 30

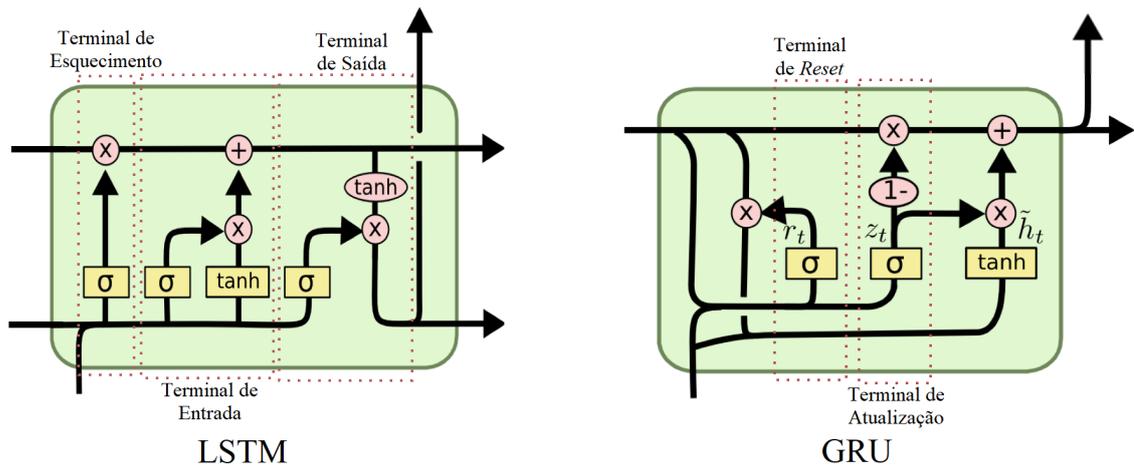


Figura 2.6 – Estrutura de uma célula LSTM e de uma célula GRU.

Fonte: Adaptado pelo autor.³

Nos blocos de LSTM/GRU presentes neste trabalho, a função de tangente hiperbólica é utilizada como função de ativação. Já que estes blocos fazem uso de contextos anteriores, e portanto de gradientes anteriores na etapa de *backpropagation*, há uma grande possibilidade de valores negativos serem recebidos, os quais seriam sempre transformados em 0 por uma função ReLU, caracterizando uma perda de informações. Como a tangente hiperbólica é simétrica em relação ao valor 0, já que os valores de saída produzidos estão entre -1 e 1, haverá uma maior chance de convergência pelo modelo (LECUN et al., 2012).

A função de tangente hiperbólica é expressa pela equação:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad (2.3)$$

onde x é um dado de entrada de uma camada e e representa o número de Euler com um valor aproximado a 2,71828.

2.1.3 Base de Dados *PhysioNet*

A base de dados utilizada nesse trabalho é pertencente à *PhysioNet* (GOLDBERGER et al., 2000) e se chama *EEG Motor Movement/Imagery Dataset* (SCHALK et al., 2004). Ela conta com 14 sessões de aquisição de dados, efetuadas em 109 indivíduos e registradas por meio do uso de 64 eletrodos, onde cada um é amostrado à 160Hz. O posicionamento destes eletrodos pode ser visualizado na Figura 2.7.

As sessões são numeradas de 1 a 14 e as tarefas realizadas pelos indivíduos consistem em:

de agosto de 2022.

⁴ Imagem disponível em: <https://physionet.org/content/eegmidb/1.0.0/64_channel_sharbrough.png>. Acesso em 9 de agosto de 2022.

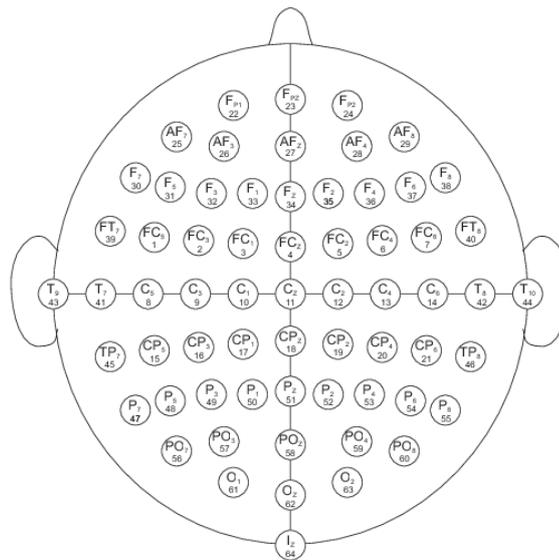


Figura 2.7 – Posicionamento dos eletrodos no couro cabeludo.

Fonte: *PhysioNet*.⁴

- *Resting Eyes Open* ou REO: O indivíduo permanece em estado de repouso com os olhos abertos (Sessão 1);
- *Resting Eyes Closed* ou REC: O indivíduo permanece em estado de repouso com os olhos fechados (Sessão 2);
- T1R1, T1R2 e T1R3: O indivíduo abre e fecha o punho correspondente ao lado da tela em que um alvo aparece (Sessões 3, 7 e 11);
- T2R1, T2R2 e T2R3: O indivíduo imagina abrir e fechar o punho correspondente ao lado da tela em que um alvo aparece (Sessões 4, 8 e 12);
- T3R1, T3R2 e T3R3: O indivíduo abre e fecha os punhos ou os pés, caso o alvo apareça em cima da tela ou embaixo, respectivamente (Sessões 5, 9 e 13);
- T4R1, T4R2 e T4R3: O indivíduo imagina abrir e fechar os punhos ou os pés, caso o alvo apareça em cima da tela ou embaixo, respectivamente (Sessões 6, 10 e 14).

As sessões 1 e 2 contam cada uma com 61 segundos de gravação ou 9760 amostras, enquanto cada uma das sessões de 3 a 14 contam com 2 minutos de gravação, ou 19200 amostras.

Na Figura 2.8 é possível verificar a variação entre os sinais de indivíduos diferentes em estado de repouso, e na Figura 2.9 é possível verificar a variação entre os sinais de um mesmo indivíduo executando tarefas diferentes. Estas variações entre os sinais presentes na base de dados demonstram que o EEG cumpre com uma característica fundamental para a construção de sistemas biométricos, a distinguibilidade, explicitada na Subseção 2.1.1.

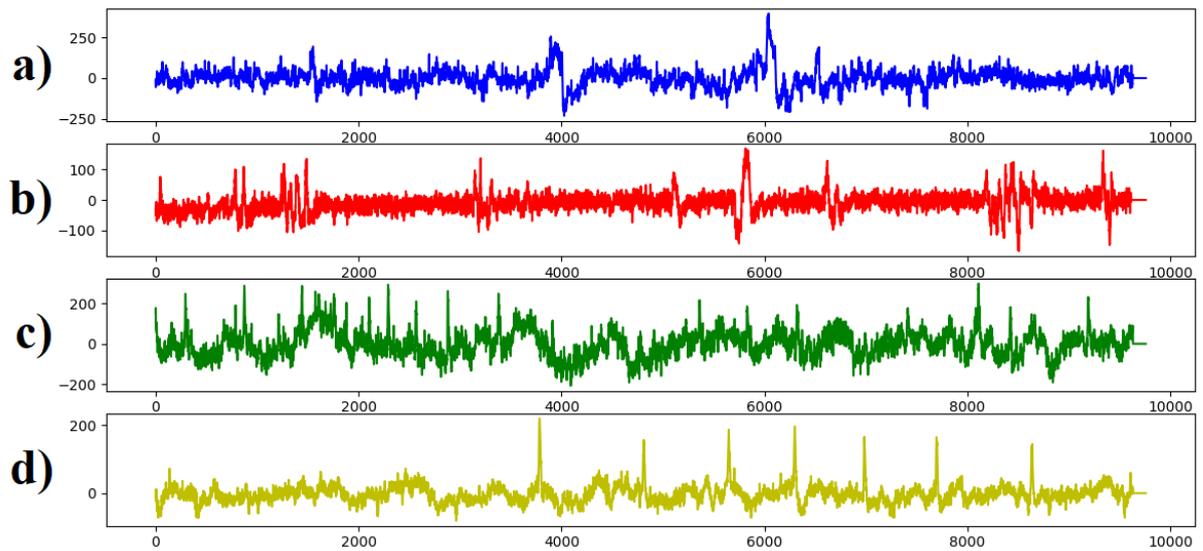


Figura 2.8 – Diferenciação entre sinais de indivíduos diferentes em estado de repouso. Os sinais a, b, c e d são referentes ao primeiro eletrodo da tarefa REO dos indivíduos 1, 2, 3 e 4 da base de dados *EEG Motor Movement/Imagery Dataset*.

Fonte: Elaborado pelo autor.

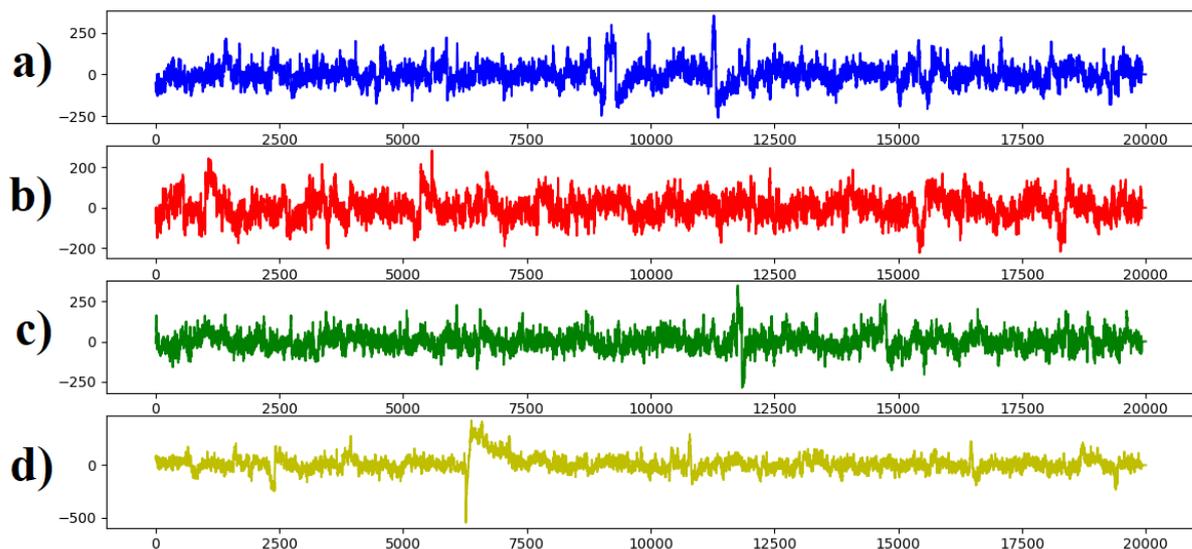


Figura 2.9 – Diferenciação entre sinais de um mesmo indivíduo executando tarefas diferentes. Os sinais a, b, c e d são referentes ao primeiro eletrodo das tarefas T1R1, T2R1, T3R1 e T4R1 do indivíduo 1 da base de dados *EEG Motor Movement/Imagery Dataset*.

Fonte: Elaborado pelo autor.

2.1.4 Filtragem de Dados

Os sinais de um eletroencefalograma podem ser classificados quanto à sua frequência, sendo as principais: delta (0.5-4Hz), teta (4-8Hz), alfa (8-12Hz), beta (12-35Hz) ou gama (35Hz ou superior). Cada um dos tipos de frequência também está atrelado a um estado de relaxamento ou concentração mental em que um indivíduo se encontra: quanto menor é a frequência de um

signal, mais relaxado ele se encontra (ABHANG; GAWALI; MEHROTRA, 2016).

No entanto, em função de diversos fatores, tais como a qualidade do gel condutor e dos eletrodos empregados na captura de sinais EEG, tais sinais são sujeitos à muitos ruídos (JAMIL; JAMIL; MAJID, 2021). Estes ruídos atrapalham na identificação de um indivíduo pela rede neural e impactam negativamente nos resultados obtidos pela mesma. Deste modo, torna-se necessário o pré-processamento destes sinais, o qual pode ser realizado através da utilização de filtros.

Os filtros utilizados em processamento de sinais podem ser classificados em *Infinite Impulse Response* (IIR) (AGRAWAL et al., 2021), os quais possuem uma infinita duração em sua resposta de impulso (saída do filtro quando um sinal curto é aplicado em sua entrada), continuando indefinidamente, e em *Finite Impulse Response* (FIR) (CHANDRA; CHATTOPADHYAY, 2016), os quais possuem uma finita duração em sua resposta de impulso, se tornando zero após alguma passagem de tempo.

A implementação de um filtro IIR pode ser definida pela equação:

$$y(n) = \sum_{i=0}^P b_i \cdot x(n-i) - \sum_{j=1}^Q a_j \cdot y(n-j), \quad (2.4)$$

onde $y(n)$ é o sinal de saída, $x(n)$ é o sinal de entrada, P é a ordem de *feedforward* do filtro, b_i são coeficientes *feedforward* do filtro, Q é a ordem de *feedback* do filtro e a_i são coeficientes *feedback* do filtro.

Pode-se entender a parte *feedback* de um filtro IIR como a parte de retroalimentação, onde um sinal de saída é levado em consideração para calcular os sinais de saída posteriores, e a parte de *feedforward* como a parte de alimentação, onde apenas o sinal de entrada é levado em consideração para o cálculo de um sinal de saída.

Já a implementação de um filtro FIR pode ser definida pela equação:

$$y(n) = \sum_{i=0}^N b_i \cdot x(n-i), \quad (2.5)$$

onde $y(n)$ é o sinal de saída, $x(n)$ é o sinal de entrada, N é a ordem do filtro e b_i é o valor da resposta de impulso no instante i do filtro.

Quanto a forma de atenuação dos sinais, os filtros ainda podem ser classificados em:

- Passa-baixa, onde todos os sinais que possuem uma frequência menor do que a um limiar a não são alterados, enquanto o restante dos sinais são atenuados;
- Passa-alta, onde todos os sinais que possuem uma frequência maior do que a um limiar a não são alterados, enquanto o restante dos sinais são atenuados;

- Passa-banda, onde todos os sinais que possuem uma frequência entre os limiares a e b não são alterados, enquanto o restante dos sinais são atenuados;
- Rejeita-banda, onde todos os sinais que possuem uma frequência menor que um limiar a ou maior que um limiar b não são alterados, sendo $a < b$, enquanto o restante dos sinais são atenuados;

A utilização de um filtro não somente pode remover ruídos e dados impuros presentes em um sinal de EEG, como também podem expor quais estados de relaxamento ou concentração podem contribuir mais na construção de um sistema biométrico, tanto para o modo de identificação quanto para o de verificação.

2.1.5 Métricas Utilizadas

A métrica utilizada para avaliar o desempenho do modelo no modo de identificação foi a acurácia, que mede a frequência com que a rede neural acerta as previsões estabelecidas para um conjunto de dados de entrada. Considerando Y como o número de previsões que a rede neural acertou e Z como o número total de previsões feitas, a acurácia é obtida por Y/Z (JAVOID et al., 2016).

A avaliação da rede neural é realizada ao utilizar os dados de teste, os quais não são utilizados em seu processo de treinamento, já que isto ocasionaria em um *overfitting* (ou sobreajuste) da rede. O sobreajuste ocorre quando um modelo neural treinado apresenta bons resultados para amostras usadas em seu treinamento, mas apresenta resultados ruins para amostras que não foram usadas durante este processo (XU; GOODACRE, 2018).

Assim, avaliar uma rede neural com base nos dados de treinamento ou validação não apresentaria resultados que servissem de comparação a resultados obtidos por outros trabalhos, ou que indicassem que a rede poderia obter um bom desempenho em situações fora do âmbito de pesquisa.

Então, no cálculo da acurácia, para cada amostra dos dados de teste, Y será incrementado em 1 se a rede reconhecer o indivíduo corretamente e Z será incrementado em 1 independentemente deste reconhecimento.

Apenas a acurácia foi reportada no modo de identificação pois o trabalho proposto em Yang, Deravi e Hoque (2018), que foi comparado com este, utilizou somente a acurácia para tal modo.

Há também uma outra métrica utilizada por trabalhos que possuem como objetivo uma alto desempenho no modo de verificação, chamada de *Equal Error Rate* (EER). Para calculá-la, devemos primeiramente estipular diversos limiares, que determinarão se dois vetores de características, extraídos dos sinais de EEG, pertencem a um mesmo indivíduo ou não.

No contexto de similaridade, quanto menor é um limiar, maior será a chance de dois vetores de características de indivíduos diferentes serem acusados como pertencentes ao mesmo indivíduo, a qual é chamada de *False Acceptance Rate* (FAR). De modo similar, quanto maior é um limiar, maior será a chance de dois vetores de características de um mesmo indivíduo serem acusados como pertencentes a indivíduos diferentes, a qual é chamada de *False Rejection Rate* (FRR). Assim, é possível definir o *Equal Error Rate* (EER) como o ponto em que o FAR é igual ao FRR (SCHONS et al., 2017), como podemos visualizar na Figura 2.10.

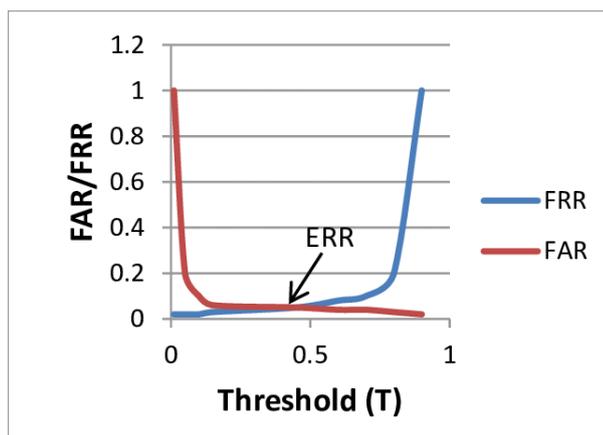


Figura 2.10 – Visualização de um EER.

Fonte: ResearchGate.⁵

O EER foi utilizado como uma métrica no presente trabalho em alguns testes iniciais cujo foco era no modo de verificação, apresentados no Capítulo 4, e como critério de desempate quando necessário.

2.2 Trabalhos Relacionados

Uma CNN e 108 dos 109 indivíduos presentes na base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004) foram utilizados em Schons et al. (2017) para construir um sistema biométrico com foco no modo de verificação. Utilizou-se 90% dos sinais gravados em estado de repouso com olhos abertos (REO - *Rest Eyes Open*) para treinamento, os 10% remanescentes para validação, e os sinais gravados em estado de repouso com olhos fechados (REC - *Rest Eyes Closed*) para teste. A métrica de *Equal Error Rate* (EER), gerada por meio de comparações entre pares genuínos (intra-classe) e impostores (inter-classe) de vetores de características, foi utilizada para medir a eficiência da rede neural. Ao utilizar um tamanho de janela de 12 segundos e um filtro passa-banda de 30-50Hz, um EER de 0,19% foi obtido.

O trabalho de Schons et al. (2017) foi utilizado como base para esta monografia, e portanto, muitas etapas da metodologia utilizada nesta serão melhor detalhadas em capítulos posteriores.

⁵ Imagem disponível em: <https://www.researchgate.net/figure/System-Equal-Error-Rate-EER_fig11_321406738>. Acesso em 07 de out. de 2021.

O trabalho proposto em Mota et al. (2021) é uma extensão do trabalho (SCHONS et al., 2017) e explora a utilização de camadas de *Squeeze-and-Excitation*, as quais têm o propósito de “melhorar a capacidade representacional de uma rede ao realizar uma recalibração dinâmica de características acerca dos canais” (HU; SHEN; SUN, 2018). Ao utilizar a CNN proposta por Schons et al. (2017), e compor os dados de treinamento e validação com a tarefa *T1R1* da base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004), e os dados de teste com a tarefa *T1R2*, um EER de 0,1% foi obtido.

Em Lai et al. (2019), utiliza-se um tamanho de janela de 1 segundo e se analisa qual é o melhor formato de entrada de um sinal EEG, gravado durante estado de repouso, para uma CNN construída para identificação biométrica. Este trabalho também utiliza os sinais gravados em estado de repouso com olhos abertos (REO) e fechados (REC) da base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004), de todos os 109 indivíduos.

A CNN utilizada por Lai et al. (2019) foi desenvolvida no trabalho de Ma et al. (2015), e escolhida por “apresentar uma alta acurácia de identificação apesar de possuir uma arquitetura rasa”. Ela é composta por duas camadas de convolução 2D, cada uma com 6 filtros e seguidas por uma camada de *pooling* 2D, e por fim uma camada densa com 109 neurônios para classificação. Diferentemente das redes neurais de outros trabalhos, a arquitetura desta rede é 2D pois os formatos de entrada explorados consistem em rearranjos da entrada original em imagens e matrizes bidimensionais. Entre diversos formatos de entrada, o sinal de EEG sem pré-processamento e sem nenhuma modificação é o formato que obteve a maior acurácia: 79,08%.

De maneira similar, os autores de Fan, Shi e Li (2021) também utilizam REO e REC relativo aos 109 indivíduos presentes na base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004) em uma CNN no cenário de identificação biométrica. A CNN em questão é composta por uma camada de *independent component analysis* (ICA), que por sua vez consiste em uma matriz W que será multiplicada pela matriz de entrada X da rede neural e será otimizada enquanto for treinada, três camadas de convolução 2D, cada uma com 32 filtros e seguidas por uma camada de *pooling* 2D, uma camada densa de 512 neurônios, uma camada de *dropout* com uma taxa de 50%, e por fim, uma camada densa com 109 neurônios para classificação.

O pré-processamento dos dados no trabalho de Fan, Shi e Li (2021) consiste apenas na normalização destes, e no deslizamento de uma janela de 0,5 segundos, com uma distância de 0,25 segundos entre uma janela e outra, para efetuar o *data augmentation*. O tamanho de janela citado também foi utilizado para o treinamento da rede neural, e um resultado de 99,32% de acurácia foi atingido.

Em Das, Maiorana e Campisi (2017), sinais EEG de 40 indivíduos foram gravados pelos próprios autores utilizando 17 eletrodos, e uma janela de 0,6 segundos. Uma tela mostrava aos indivíduos formas geométricas selecionadas aleatoriamente, e estes foram instruídos a se concentrar apenas quando um círculo aparecesse, sendo este o alvo, e ignorar a aparição de outras formas geométricas, sendo estas não-alvo. A CNN utilizada é composta por três camadas

de convolução, onde há uma camada de *pooling* após a primeira e segunda camadas, e uma camada densa com 40 neurônios. As camadas de convolução possuem 77, 320 e 1024 filtros, respectivamente.

A etapa de pré-processamento dos dados de Das, Maiorana e Campisi (2017) consiste em aplicar um filtro *common average referencing* (CAR), um filtro passa-banda de 0,5-8Hz, um *downsampling* do sinal de 256Hz para 128Hz, uma normalização, e por fim, um processo de *detrending*, onde cada sinal é subtraído por uma linha que mais se ajusta ao sinal original (*best-fit line*). Ao classificar os indivíduos no modo biométrico de identificação, foi reportado 98,8% de acurácia em comparações não-alvo vs não-alvo (quando tanto a amostra do sinal EEG quanto a saída da CNN indicaram que o indivíduo estava ignorando um não-alvo) e 80,65% de acurácia em comparações alvo vs alvo.

Um sistema baseado em coeficientes de *wavelets* foi utilizado em Yang, Deravi e Hoque (2018) para a extração de características dos 109 indivíduos presentes na base de dados *PhysioNet*. Foram utilizados 9 dos 64 eletrodos disponíveis, julgados como os que podem apresentar informações de maior capacidade de diferenciação entre indivíduos. O trabalho também estabeleceu três protocolos para avaliação:

- Protocolo P1: qual(is) região(ões) cerebral(is) apresenta(m) melhores resultados;
- Protocolo P2: qual combinação entre dados de treinamento e dados de teste apresentam melhores resultados;
- Protocolo P3: o quanto os resultados pode melhorar ao manter os mesmos dados de teste e adicionar mais dados de treinamento, provenientes de tarefas diferentes.

Yang, Deravi e Hoque (2018) concluíram que quanto mais regiões cerebrais forem utilizadas para treinamento e teste do sistema, melhores serão os resultados. Também foi observado que dados de treinamento e teste provenientes de diferentes tarefas realizadas pelos indivíduos não implicam na piora dos resultados, e que quanto mais dados de treinamento forem utilizados, melhores serão os resultados obtidos pela rede, obedecendo a Lei dos Rendimentos Decrescentes (SHEPHARD; FÄRE, 1974).

Os protocolos P1, P2 e P3 mencionados também foram abordados no presente trabalho, e os resultados estão presentes na Subseção 4.4.

Já em Jijomon e Vinod (2021) utilizou-se *auditory evoked potentials* (estímulos auditivos) em 20 indivíduos, 2 eletrodos, amostrados à 500Hz, localizados na parte frontal do cérebro para capturar os sinais de EEG e um tamanho de janela de 1 segundo. A coleta dos dados foi feita pelos próprios autores. Para o cenário de identificação de um sistema biométrico, foi utilizada uma rede neural composta por duas camadas de convolução, cada uma com 36 filtros de tamanho 50, seguidos de uma camada de LSTM com 50 unidades, e por fim, uma camada densa com 50

neurônios. Ao pré-processar os dados utilizando um filtro passa-banda de 8-100Hz, normalizá-los e aplicar um processo de *downsampling* por um fator de dois (ou seja, apenas uma a cada duas amostras eram mantidas), um resultado de 99,53% de acurácia foi atingido.

Em Svetlakov et al. (2022), semelhante a trabalhos citados anteriormente, foi utilizado sinais REO e REC dos 109 indivíduos presentes na base de dados EEG *Motor Movement/Imagery Dataset* (SCHALK et al., 2004), de 8 dos 64 eletrodos disponíveis, como forma de reduzir o custo computacional. Para obter os espectrogramas dos sinais de EEG, foi utilizado a Transformada de Hilbert-Huang, dentre outras operações, e uma posterior normalização foi utilizada na etapa de pré-processamento dos dados.

A CNN utilizada por Svetlakov et al. (2022) possui sete camadas de convolução, cada uma seguida por uma camada de *batch normalization*. Há uma camada de *dropout*, com uma taxa de 70%, anterior à primeira e quinta camadas de convolução, e duas camadas densas após a última camada de *batch normalization*, com 256 e 128 neurônios, respectivamente. Para o modo de verificação, um resultado de 14,63% de EER foi obtido.

Em Oord et al. (2016) é explorado a utilização de um modelo neural baseado em convoluções causais dilatadas para a geração de áudio, como fala humana e música. O modelo não utiliza camadas de *pooling*, e sua saída possui a mesma dimensionalidade de tempo de sua entrada. Foram conduzidos quatro testes no modelo, sendo estes: geração de fala, texto para fala, musical e reconhecimento de fala.

No primeiro teste, o modelo WaveNet foi capaz de capturar e reproduzir as características de fala de 109 indivíduos presentes na base de dados CSTR voice cloning toolkit (VCTK). No segundo teste, o modelo foi treinado com as bases de dados de inglês norte americano e chinês mandarim utilizadas pelo Google para a construção de um sistema texto para fala, e obteve uma pontuação média de opinião de 4, equivalente a uma boa performance. Já no terceiro teste, o modelo foi treinado com as bases de dados MagnaTagATune e YouTube piano, que contam com 200 horas e 60 horas de áudio musical, respectivamente, e as amostras produzidas pelo modelo foram julgadas subjetivamente como “harmônico e esteticamente agradável muitas das vezes”. No último teste conduzido, camadas de *pooling* foram adicionadas após as camadas de convolução do modelo e a base de dados TIMIT foi utilizada em seu treinamento, resultando em um *phone error rate* (PER) de 18.8.

Alguns dos autores citados anteriormente utilizam CNNs, como Schons et al. (2017), para a construção de sistemas biométricos baseados em EEG. Já outros utilizam de LSTMs para tal, como em Jijomon e Vinod (2021).

O presente trabalho então busca utilizar as arquiteturas CNN e LSTM/GRU, bem como a unificação de tais arquiteturas, além de técnicas de pré-processamento e *data augmentation*, na tentativa de melhorar os resultados obtidos pelo modelo para o contexto de identificação biométrica, e utilizar os EEGs dos 109 indivíduos presentes na base de dados PhysioNet EEG

Motor Movement/Imagery Dataset. Como forma de avaliação do modelo, foram realizados os testes presentes nos protocolos P1, P2 e P3 propostos por Yang, Deravi e Hoque (2018), com todos os 64 eletrodos disponíveis, bem como os 9 eletrodos utilizados por Yang, Deravi e Hoque (2018), para uma comparação mais justa. Também foi analisado qual é o impacto da utilização de *padding* causal nas camadas de convolução do modelo, de modo a replicar as convoluções causais dilatadas usadas em Oord et al. (2016).

3 Desenvolvimento

Neste capítulo são apresentados os métodos utilizados para o processamento dos sinais de EEG e construção da rede neural utilizada. A metodologia do trabalho proposto seguiu o *workflow* apresentado na Figura 3.1.

Após as tarefas para treinamento e teste da rede neural serem selecionadas na base de dados, os sinais de EEG que as compõe serão pré-processados conforme o processo explicitado na Seção 3.1 (**Pré-Processamento**). Os dados das tarefas de treinamento então passarão por um processo de *data augmentation*, explicado na Seção 3.2 (**Data Augmentation**), e serão separados em dois conjuntos conforme apresentado na Seção 3.3 (**Composição dos Dados de Validação**), sendo um deles destinado para os dados de validação. As arquiteturas das redes neurais utilizadas inicialmente serão aprofundadas na Seção 3.4 (**Arquiteturas Utilizadas**) e o processo de treinamento destas fará uso dos dados de treinamento e de validação, a qual será posteriormente avaliada utilizando os dados de teste conforme demonstra a Seção 3.5 (**Avaliação da Rede**).

Redes estado da arte, ou seja, aquelas que apresentam bons resultados para as tarefas as quais foram designadas, e são referências para o estado atual do campo de pesquisa, são treinadas normalmente para imagens (2D), como propostas de modificações às arquiteturas da ImageNet (DENG et al., 2009) e ResNet (HE et al., 2015), por exemplo. Como este trabalho utiliza dados 1D, não há uma variedade de modelos treinados o que dificulta o processo de transferência de aprendizado, no qual o conjunto de pesos de uma rede pré-treinada pode ser transferido a uma nova rede neural.

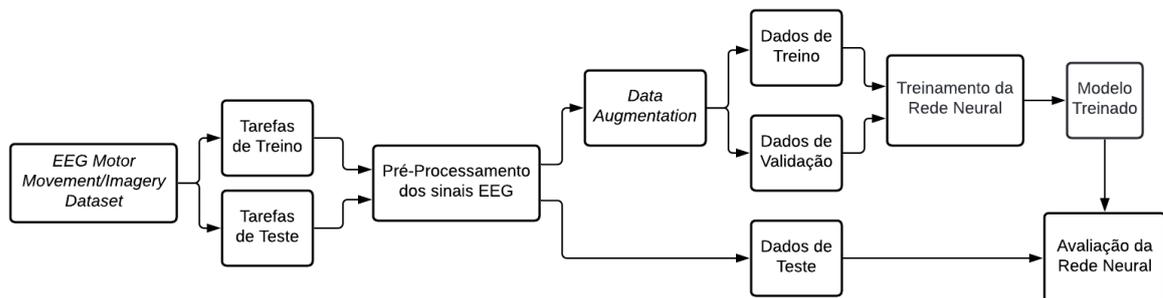


Figura 3.1 – Metodologia desenvolvida no trabalho proposto.

Fonte: Elaborado pelo autor.

3.1 Pré-Processamento

Como dito anteriormente na Subseção 2.1.4, os sinais de EEG são sujeitos à muitos ruídos (JAMIL; JAMIL; MAJID, 2021), sendo necessário realizar um pré-processamento destes sinais para melhorar o desempenho da rede neural. A primeira etapa de pré-processamento dos dados, proposta por Fraschini et al. (2014), “consiste em aplicar filtros passa-banda aos sinais crus de EEG”, sem a realização prévia de limpeza destes dados, como detecção e correção de certas inconsistências.

Deste modo, o presente trabalho faz uso de filtros passa-banda na etapa de pré-processamento dos sinais de EEG, primeiramente propostos por Fraschini et al. (2014) e também utilizados por Schons et al. (2017), o qual foi utilizado como referência, para melhorar a qualidade dos sinais de EEG e dos dados obtidos por meio deles. Os filtros propostos foram os de 1-50Hz (delta a gama), 10-30Hz (alfa a beta) e 30-50Hz (beta a gama), e foram implementados de duas maneiras diferentes, utilizando as funções *sosfilt* e *filtfilt* do módulo *scipy* da linguagem *Python*, que utilizam filtros IIR e FIR nos dados de um sinal, respectivamente.

A Figura 3.2 mostra um exemplo de sinal de EEG sem e com a aplicação de um filtro passa-banda (em azul e vermelho, respectivamente).

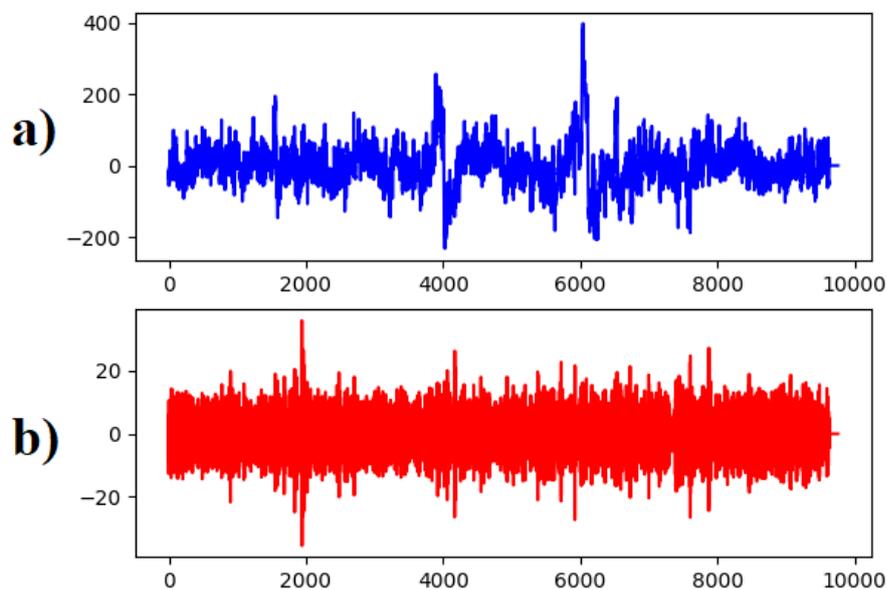


Figura 3.2 – Diferenciação entre um sinal não filtrado e um filtrado. a) Sinal de EEG captado pelo primeiro eletrodo do primeiro indivíduo presente na base de dados *EEG Motor Movement/Imagery Dataset*, utilizando a tarefa REO; b) Sinal **a** após a aplicação de um filtro FIR passa-banda de 30-50 Hz e de ordem 12.

Fonte: Elaborado pelo autor.

3.2 Data Augmentation

Uma rede neural utilizará de dados de entrada para poder aprender a tarefa a qual foi designada. Desta forma, redes neurais convolutivas que possuam um grande número de camadas necessitarão de uma grande quantidade de dados para ajustar os pesos presentes em cada uma delas.

O processo de *data augmentation* proposto por Schons et al. (2017) para aumentar o número de dados de entrada foi o de deslizar uma janela de 12 segundos, equivalente à 1920 amostras (já que os eletrodos utilizados na captura dos EEGs são amostrados à 160Hz) pelo sinal completo. Tal valor foi escolhido pois o trabalho de Fraschini et al. (2014) foi utilizado como método base para um protocolo de avaliação.

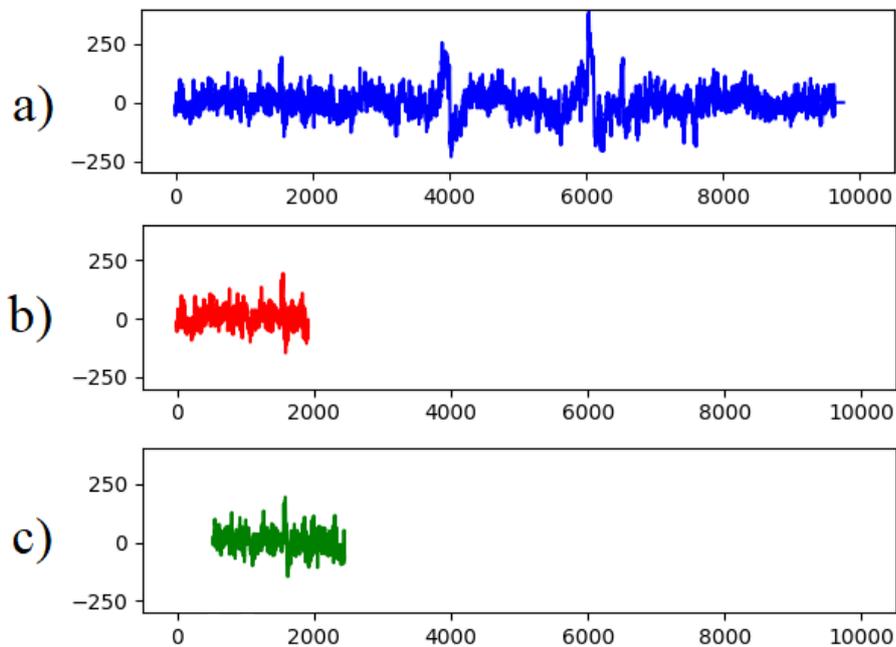


Figura 3.3 – *Data augmentation* com janela deslizante. (a) Sinal de EEG captado pelo primeiro eletrodo do primeiro indivíduo presente na base de dados *EEG Motor Movement/Imagery Dataset*, utilizando a tarefa REO; (b) Recorte de 1920 amostras efetuado em (a), partindo do início do sinal. (c) Recorte de 1920 amostras efetuado em (a), após a janela ter deslizado 500 amostras.

Fonte: Elaborado pelo autor.

Conforme apresentado pela Figura 3.3, o processo de *data augmentation* pode ser descrito nos seguintes passos:

- Partindo do início do sinal completo, recortamos 1920 amostras deste sinal para compor uma amostra para a CNN;
- A janela é então deslocada por um número fixo de amostras, chamado de *offset*, e outro recorte é efetuado;

- Os passos descritos anteriormente são repetidos até o final do sinal completo ser atingido;
- Se ao deslizar a janela pela última vez (quando o final do sinal é atingido), ocorre o recorte de uma janela de tamanho diferente das outras geradas até então, esta janela é descartada.

Este processo foi então implementado com o tamanho da janela igual a 1920 amostras, já que o trabalho proposto utiliza o trabalho de Schons et al. (2017) como referência, e o valor de *offset* será avaliado e determinado em experimentos posteriores.

3.3 Composição dos Dados de Validação

Durante o processo de treinamento de uma rede neural, um conjunto de dados, denominado de validação, será utilizado para avaliar os resultados desta rede e atualizar seus pesos. No presente trabalho, parte dos dados de treinamento são destinados a compor os dados de validação. Foi então determinado que 10% dos dados de treinamento iriam compô-los, assim como demonstra a Figura 3.4, valor este escolhido por Schons et al. (2017).

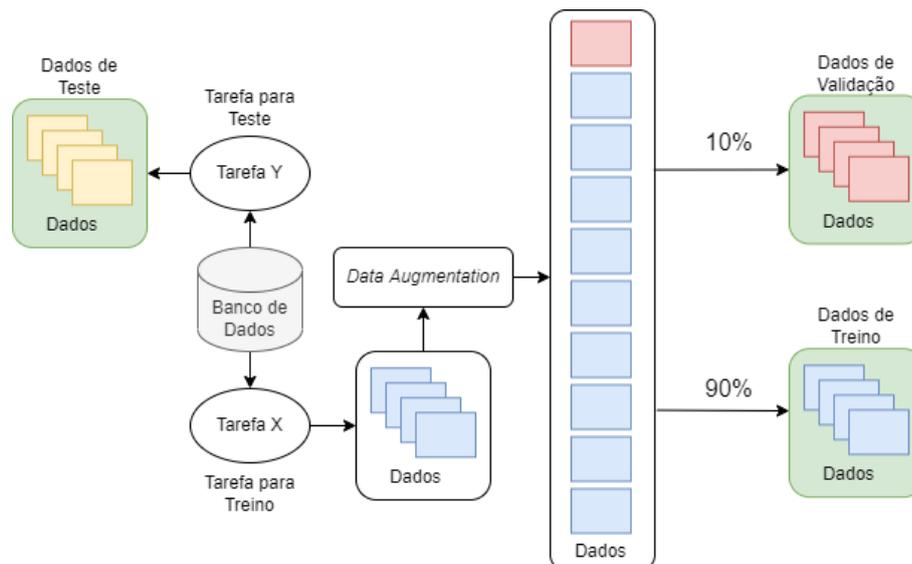


Figura 3.4 – Modo de composição das tarefas de treino, validação e teste utilizado no trabalho proposto.

Fonte: Elaborado pelo autor.

3.4 Arquiteturas Utilizadas

Uma das arquiteturas implementadas inicialmente foi baseada na CNN utilizada por Schons et al. (2017), podendo ser observada na Tabela 3.1. Vale lembrar que o formato da entrada da primeira camada da CNN (Conv1) é 1920 x 64 devido ao tamanho da janela utilizado e ao número de eletrodos empregados na gravação dos sinais EEG, onde cada eletrodo corresponde a um canal do sinal.

Nome	Tipo	Formato da Entrada	# Filtros / Unidades	Tamanho do Filtro	Stride	Ativação
Conv1	<i>Convolution</i>	1920 x 64	96	11	1	ReLU
Norm1	<i>Batch Norm.</i>	1910 x 96	N/A	N/A	N/A	Não
Pool1	<i>Max Pooling</i>	1910 x 96	N/A	2	4	Não
Conv2	<i>Convolution</i>	478 x 96	128	9	1	ReLU
Norm2	<i>Batch Norm.</i>	470 x 128	N/A	N/A	N/A	Não
Pool2	<i>Max Pooling</i>	470 x 128	N/A	2	2	Não
Conv3	<i>Convolution</i>	235 x 128	256	9	1	ReLU
Norm3	<i>Batch Norm.</i>	227 x 256	N/A	N/A	N/A	Não
Pool3	<i>Max Pooling</i>	227 x 256	N/A	2	2	Não
FC1	<i>Dense</i>	28928	4096	N/A	N/A	ReLU
FC2	<i>Dense</i>	4096	4096	N/A	N/A	ReLU
FC3	<i>Dense</i>	4096	256	N/A	N/A	Não
Norm4	<i>Batch Norm.</i>	256	N/A	N/A	N/A	Não
Drop	<i>Dropout</i>	256	N/A	N/A	N/A	Não
FC4	<i>Dense</i>	256	109	N/A	N/A	<i>Softmax</i>

Tabela 3.1 – Arquitetura da CNN utilizada inicialmente.

A CNN é composta por três sequências de camadas de convolução, seguidas de *batch normalization* e *max pooling*, três camadas densas, um *batch normalization*, uma camada de *dropout*, configurada com uma taxa de 10%, e finalmente uma camada densa que utiliza uma função de ativação *softmax* para classificar os indivíduos. A função de ativação utilizada nas camadas de convolução e em algumas das camadas densas foi a ReLU.

O outro tipo de arquitetura implementada inicialmente baseou-se na utilização de blocos de LSTM ou GRU contíguos, e pode ser visualizada na Tabela 3.2. A rede é composta por cinco blocos de LSTM ou GRU, com 10 unidades por bloco, seguidos de duas camadas densas, onde a última destas utiliza uma função de ativação *softmax* para classificar os indivíduos. A função de ativação utilizada pelos blocos de LSTM/GRU foi a da tangente hiperbólica, ou tanh.

Nome	Tipo	Formato da Entrada	# Unidades	Ativação
Block1	LSTM/GRU	1920 x 64	10	tanh
Block2	LSTM/GRU	1920 x 10	10	tanh
Block3	LSTM/GRU	1920 x 10	10	tanh
Block4	LSTM/GRU	1920 x 10	10	tanh
Block5	LSTM/GRU	1920 x 10	10	tanh
FC1	<i>Dense</i>	19200	256	Não
FC2	<i>Dense</i>	256	109	<i>Softmax</i>

Tabela 3.2 – Arquitetura que utiliza LSTM/GRU utilizada inicialmente.

Vale ressaltar que o objetivo era avaliar os resultados obtidos através da utilização de blocos de LSTM e GRU em vez de encontrar uma arquitetura baseada em tais blocos, e que a

arquitetura apresentada se mantém quando testes utilizando blocos bidirecionais de LSMT ou GRU são feitos.

Os valores para o número de blocos de LSTM/GRU e o número de unidades por bloco foram definidos empiricamente. Experimentos para a definição do melhor posicionamento destes blocos de LSTM/GRU em relação à arquitetura da CNN apresentada anteriormente, bem como valores para o número de blocos e número de unidades por bloco serão realizados na continuação deste trabalho.

Para avaliar o uso dos 9 eletrodos selecionados no trabalho de [Yang, Deravi e Hoque \(2018\)](#), também foi construída uma rede que possui um formato de entrada da primeira camada de convolução (Conv1) de 1920 x 9, e mantém todas as outras características apresentadas na Tabela 3.1.

3.5 Avaliação da Rede

No processo de avaliação dos resultados de uma rede neural no modo de identificação, foi utilizado a acurácia (também mencionada como *Rank-1 Accuracy* em alguns trabalhos) como métrica. Esta é dada em porcentagem, e quanto mais próxima de 100%, melhores são os resultados obtidos pela rede.

Como este trabalho é baseado no trabalho proposto por [Schons et al. \(2017\)](#), primeiro avaliou-se a quantidade de épocas de treinamento no modo de verificação, para depois adaptar para o modo de identificação biométrica. O modelo neural é avaliado no modo de verificação após os resultados obtidos no modo de identificação, e logo, após a etapa de treinamento, assim como pode ser observado na Figura 3.1.

Para a avaliação dos resultados no modo de verificação, realizada em alguns testes presentes no Capítulo 4, os quais utilizam apenas a CNN apresentada, foram removidas as duas últimas camadas (Drop e FC4) a fim de obter vetores de características, e então compará-los com base em diferentes limiares estabelecidos, gerando pares genuínos (quando estes vetores são semelhantes, e considerados pertencentes ao mesmo indivíduo) e impostores (quando estes vetores são suficientemente diferentes, e considerados pertencentes a indivíduos diferentes). Estes pares são avaliados por meio de distância euclidiana, e a métrica de *Equal Error Rate* (EER) é calculada, a qual é dada em porcentagem, e quanto mais próxima de 0%, melhores são os resultados obtidos pela rede.

4 Experimentos e Resultados

Neste capítulo são apresentados os resultados dos experimentos realizados e discussões acerca deles. Os experimentos foram conduzidos em uma máquina com um processador Intel i9-10900 @ 2.80GHz, 128GB de RAM DDR4 e placa de vídeo RTX 3090 com 24GB de memória GDDR6X.

As redes neurais utilizadas foram treinadas utilizando *Stochastic Gradient Descent* com um coeficiente de *momentum* igual à 0.9 como o algoritmo de otimização, e o tamanho dos *mini-batches* foi definido como 100. Inicialmente, o número de épocas definido foi 60. Seguindo o que foi proposto por Schons et al. (2017), o *learning rate* empregado possui o valor 0,1 durante as primeiras 2 épocas, 0,01 das épocas de 3 a 37 e 0,001 nas épocas restantes. Desta forma, o *learning rate* é diminuído conforme o número de épocas na tentativa de obter um conjunto ótimo de pesos da rede neural (BENGIO, 2012).

Todos os experimentos presentes neste capítulo tiveram os dados de treinamento e validação compostos por REO e os dados de teste compostos por REC. Os experimentos descritos nas Seções 4.1 e 4.2 utilizaram um modelo neural com a arquitetura CNN, enquanto os descritos na Seção 4.3 utilizaram um modelo neural com a arquitetura que utiliza blocos LSTM/GRU.

Os experimentos cujos resultados estão presentes desde a Seção 4.1 (**Filtragem e Normalização**) até a seção Seção 4.3 (**Utilização de Blocos LSTM e GRU**) tiveram os dados de treinamento e validação compostos por REO e os dados de teste compostos por REC.

4.1 Filtragem e Normalização

Foram realizados testes com os três filtros passa-banda mencionados na Seção 3.1, usados por Schons et al. (2017), sendo estes 1-50Hz, 10-30Hz e 30-50Hz, os quais exploram os sinais nas faixas de frequência delta a gama, alfa a beta e beta a gama, respectivamente. Tais filtros possuem ordem 12 e foram implementados utilizando as funções *sosfilt* e *filtfilt* do módulo *scipy* da linguagem *Python*, as quais aludem à filtros IIR e FIR, respectivamente.

Na Tabela 4.1 encontra-se a acurácia para cada um destes filtros passa-banda. Também foram realizados testes utilizando a normalização dos dados após a etapa de filtragem, proposta por Schons et al. (2017), onde estes são subtraídos por sua média, depois somados com o absoluto do valor mínimo presente neles, então divididos pelo seu desvio padrão, e por fim, divididos pelo valor máximo presente neles. Tal normalização foi aplicada tanto nos dados presentes em cada canal quanto a todos os canais simultaneamente, ou seja, a informação presente em cada canal do sinal EEG foi levada em consideração para a normalização de todos os outros.

Pode-se observar que os melhores resultados foram obtidos pelo modelo ao utilizar o

Filtro do Pré-Processamento	Acc-F (%)		Acc-FN1 (%)		Acc-FN2 (%)	
	IIR	FIR	IIR	FIR	IIR	FIR
1-50Hz	49,17	50,64	1,28	5,50	13,94	22,94
10-30Hz	44,95	44,22	2,20	3,67	10,64	14,49
30-50Hz	63,30	64,04	1,47	5,69	20,37	27,52

Tabela 4.1 – Filtro e normalização no pré-processamento dos dados. Acc-F = Acurácia sem normalização; Acc-FN1 = Acurácia com normalização aplicada em cada canal separadamente; Acc-FN2 = Acurácia com normalização aplicada em todos os canais simultaneamente.

filtro FIR passa-banda de 30-50Hz (implementado com a função *filtfilt*), sem normalização dos dados. Também pode ser observado que a utilização da normalização proposta por Schons et al. (2017) afetou negativamente os resultados obtidos pelo modelo no modo de identificação, diminuindo notavelmente a acurácia independentemente do modo em que tal normalização foi aplicada.

Portanto, a utilização do filtro passa-banda 30-50Hz foi realizada na etapa de pré-processamento dos dados para todos os testes descritos posteriormente, enquanto este tipo de normalização não o foi. As únicas exceções são os teste relacionados à definição do valor de *offset* para o processo de *data augmentation*, o qual utilizou a normalização proposta por Schons et al. (2017).

4.2 Avaliação dos Parâmetros

Em um primeiro momento, o presente trabalho tentou replicar os resultados obtidos por Schons et al. (2017), cujo foco era um alto desempenho no modo de verificação. Durante esta etapa, também foi definido um valor para o *offset* do processo de *data augmentation*, o qual define o número de amostras em que a janela será deslocada entre dois recortes dos dados de treinamento ou validação.

A Tabela 4.2 apresenta os valores de acurácia e EER obtidos ao utilizar diferentes *offsets* para o processo de *data augmentation*, onde a CNN é treinada por 60 épocas, além do processo de normalização proposto por Schons et al. (2017) ser aplicado a cada canal separadamente. Apesar do melhor resultado no modo de verificação ter sido obtido ao utilizar um *offset* de tamanho 80, um *offset* de tamanho 35 (ou aproximadamente 0,22 segundos) obteve um EER apenas 0,02% menor, enquanto geraria cerca de 2,2 vezes mais amostras para o treinamento da rede. Deste modo, o tamanho de *offset* foi definido como 35 para todos os experimentos posteriores.

Após um sucesso ao tentar replicar os resultados obtidos por Schons et al. (2017) no modo de verificação (0,19% de EER), e com o valor de *offset* definido, buscou-se também definir um valor para o número de épocas em que o modelo seria treinado.

Na Tabela 4.3, que contém os resultados de treinar o modelo, sem normalização nos

<i>Offset</i>	Acurácia (%)	EER (%)
35	2,02	0,092
40	3,12	0,298
60	1,47	0,092
80	1,83	0,072
100	1,83	0,097
200	1,10	0,092
300	1,28	0,550
400	1,28	0,184

Tabela 4.2 – Tamanhos de *offset* para o processo de *data augmentation*. O experimento foi realizado no contexto de verificação biométrica.

dados, podemos visualizar que o melhor resultado para o modo de identificação é obtido ao treinar o modelo por 40 épocas, possuindo a mesma acurácia e um menor EER que ao treinar o modelo por 30 épocas, apesar da baixa variação entre estes valores.

# Epochs	Acurácia (%)	EER (%)
10	63,12	3,39
30	64,22	3,85
40	64,22	3,49
60	63,30	3,48

Tabela 4.3 – Valores de épocas em que o modelo é treinado.

4.3 Utilização de Blocos LSTM e GRU

A Tabela 4.4 apresenta os resultados de uma rede neural que utiliza 5 blocos de LSTM/GRU, descrita anteriormente na Seção 3.4, podendo ser estes bidirecionais ou não. Cada bloco possui 10 unidades, e os experimentos para determinar o número de blocos, assim como o número de unidades por bloco que resultam em uma maior acurácia, foram realizados posteriormente. A utilização de blocos LSTM apresenta uma maior acurácia do que a de blocos bidirecionais LSTM, enquanto a utilização de blocos bidirecionais GRU apresenta uma maior acurácia do que blocos GRU. Este resultado foi levado em consideração para os experimentos posteriores.

Arquitetura	Acurácia (%)
LSTM	57,25
<i>Bidirectional</i> LSTM	53,03
GRU	64,77
<i>Bidirectional</i> GRU	67,34

Tabela 4.4 – Utilizando blocos de LSTM e GRU contíguos.

A Tabela 4.5 apresenta os resultados de unir 5 blocos de LSTM ou GRU bidirecionais contíguos, com 10 unidades cada, à arquitetura da CNN apresentada na Tabela 3.1. Também foi observado os resultados da aplicação de uma outra normalização dos dados, presente em (SUN; LO; LO, 2019), após a etapa de filtragem, onde os dados presentes em cada canal foram subtraídos por sua média e depois divididos pelo seu desvio padrão.

Tanto na utilização de blocos de LSTM quanto na de blocos GRU bidirecionais, a maior acurácia foi obtida ao posicioná-los antes da primeira camada de convolução (Conv1). É notável também que a normalização de dados descrita não deteriorou a acurácia como a de (SCHONS et al., 2017), e deste modo, foi realizada em todos os experimentos posteriores.

Arquitetura	Posicionamento	Normalização	Acurácia (%)	EER (%)
LSTM	Entre Pool3 e FC1		40,73	25,51
	Anterior a Conv1		73,39	48,77
	Anterior a Conv1	✓	73,39	45,97
GRU Bidirecional	Entre Pool3 e FC1		58,35	19,00
	Anterior a Conv1		71,38	45,90
	Anterior a Conv1	✓	80,37	47,92

Tabela 4.5 – Utilizando blocos de LSTM e GRU bidirecionais, unidos à arquitetura da CNN apresentada anteriormente.

Na Tabela 4.6 se encontram os resultados obtidos ao utilizar 5 blocos de LSTM ou GRU contíguos, posicionados anteriormente a primeira camada de convolução da CNN, e variar o número de unidades por bloco. Para ambos os tipos de bloco, a maior acurácia foi obtida ao utilizar 128 unidades por bloco. Porém, o resultado obtido pelos blocos de LSTM (95,23%) é consideravelmente maior que o obtido pelos blocos de GRU bidirecionais (85,69%), e portanto, apenas a utilização dos blocos de LSTM foi considerada em experimentos posteriores.

Arquitetura	# Unidades por bloco	Acurácia (%)
LSTM	16	90,64
	32	92,29
	64	94,31
	128	95,23
	256	94,68
GRU Bidirecional	16	74,86
	32	81,10
	64	85,14
	128	85,69
	256	82,75

Tabela 4.6 – Variando o número de unidades por bloco de LSTM e GRU bidirecional.

Os resultados obtidos ao variar o número de blocos de LSTM que se encontram anterior a primeira camada de convolução da CNN, onde cada bloco possui 128 unidades, se encontram na

Tabela 4.7. A mesma acurácia, de 95,23%, foi obtida ao utilizar tanto 5 blocos, quanto 10 blocos de LSTM. Entretanto, a utilização de um menor número de blocos é preferível, já que ocasionará em um menor número de parâmetros a serem ajustados pela rede neural, e conseqüentemente, em um treinamento mais rápido. Dado os altos valores de EER, os modelos não possuem bons resultados no modo de verificação, e portanto, a diferença de 0,84% de EER não é grande o suficiente para ser considerada um critério de desempate neste caso.

# Blocos	Acurácia (%)	EER (%)
4	94,68	48,02
5	95,23	47,63
6	94,49	45,80
10	95,23	46,79

Tabela 4.7 – Variando o número de blocos de LSTM.

Diante os experimentos realizados, a arquitetura final adotada foi a de 5 blocos de LSTM, com 128 unidades cada, seguidos da CNN apresentada na Tabela 3.1, usada em (SCHONS et al., 2017). Esta arquitetura pode ser visualizada na Figura 4.1.

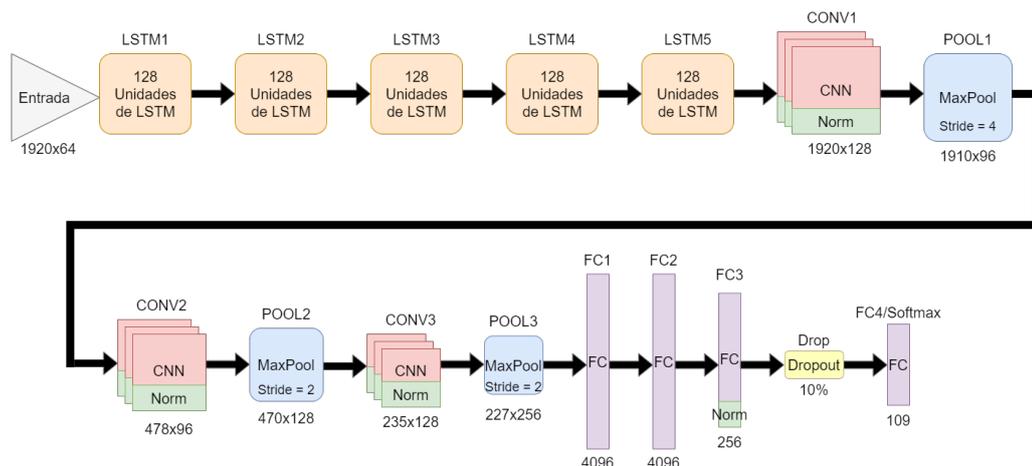


Figura 4.1 – Arquitetura final adotada. "Norm" indica *Batch Normalization*. As camadas Conv1, Conv2 e Conv3 possuem um número de filtros igual a 96, 128 e 256, com tamanhos iguais a 11, 9 e 9, respectivamente. As camadas de *MaxPooling* possuem filtros de tamanho 2 sem *padding*. As camadas Conv1, Conv2, Conv3, FC1 e FC2 possuem funções de ativação ReLU.

Fonte: Elaborado pelo autor.

4.4 Protocolos P1, P2 e P3

Experimentos relativos aos protocolos P1, P2 e P3 presentes em (YANG; DERAVID; HOQUE, 2018), usando o modelo apresentado na Figura 4.1 com todos os 64 eletrodos disponíveis e os mesmos 9 eletrodos usados por Yang, Deravi e Hoque (2018), foram realizados, tornando a comparação entre os dois trabalhos mais justa.

Na Tabela 4.8 estão presentes os resultados referentes ao protocolo P1. Pode-se observar que a maior acurácia obtida ao utilizar 9 eletrodos foi de 80,72%, ao compor os dados de treino com T4R1O e T4R3O e os dados de teste com T4R2O, capturados por eletrodos presentes na região occipital do cérebro. A maior acurácia obtida ao utilizar 64 eletrodos foi de 99,45%. ao compor os dados de treino com T3R1 e T3R3 e os dados de teste com T3R2.

Tendo em vista o resultado de 96% obtido por Yang, Deravi e Hoque (2018) no protocolo P1, podemos afirmar que o presente trabalho obteve um melhor resultado ao utilizar 64 eletrodos e um pior resultado ao utilizar os mesmos 9 eletrodos.

Tarefas de treino	Teste	Acc-9 (%)	Acc-64 (%)
T1R1F + T1R3F	T1R2F	54,61	—
T1R1M + T1R3M	T1R2M	61,53	—
T1R1O + T1R3O	T1R2O	78,50	—
T1R1 + T1R3	T1R2	65,13	98,34
T2R1F + T2R3F	T2R2F	54,53	—
T2R1M + T2R3M	T2R2M	64,79	—
T2R1O + T2R3O	T2R2O	79,85	—
T2R1 + T2R3	T2R2	66,27	98,80
T3R1F + T3R3F	T3R2F	41,33	—
T3R1M + T3R3M	T3R2M	62,73	—
T3R1O + T3R3O	T3R2O	79,52	—
T3R1 + T3R3	T3R2	64,85	99,45
T4R1F + T4R3F	T4R2F	51,01	—
T4R1M + T4R3M	T4R2M	65,68	—
T4R1O + T4R3O	T4R2O	80,72	—
T4R1 + T4R3	T4R2	70,11	98,89

Tabela 4.8 – Resultado para o protocolo P1: Acc-9 = Acurácia com 9 eletrodos; Acc-64 = Acurácia com 64 eletrodos. Modo de identificação.

Da mesma forma, a Tabela 4.9 apresenta os resultados do protocolo P2. Foi obtida uma acurácia de 99,63% ao usar T1R1 e T1R3 para os dados de treinamento e validação e T3R2 para os dados de teste, usando todos os 64 eletrodos disponíveis. Já ao utilizar apenas 9 eletrodos, o melhor resultado foi alcançado ao compor os dados de treinamento com as tarefas T4R1 e T4R3 e os dados de teste com a tarefa T2R2, resultando em uma acurácia de 73,84%.

Mais uma vez, a utilização de todos os 64 eletrodos disponíveis obteve um desempenho melhor que o trabalho de Yang, Deravi e Hoque (2018), que obteve uma acurácia de 96,15% no protocolo P2, e a utilização dos mesmos 9 eletrodos possuiu um desempenho pior.

Por fim, a Tabela 4.10 apresenta os resultados do protocolo P3, que avalia diferentes composições de dados de treinamento e sempre compõe os dados de teste com a tarefa T1R2. Foi obtida uma acurácia de 99,72% ao usar T1R1, T1R3, T2R1, T2R2, T2R3, T3R1 e T4R1 para os dados de treinamento e validação, e todos os 64 eletrodos disponíveis. Para os mesmos 9

Treino	Teste	Acc-9 (%)	Acc-64 (%)
T1R1 + T1R3	T1R2	65,13	98,34
	T2R2	60,54	98,89
	T3R2	65,04	99,63
	T4R2	65,22	99,26
	REO	54,13	95,05
	REC	32,66	91,93
T2R1 + T2R3	T1R2	63,01	98,06
	T2R2	66,27	98,80
	T3R2	52,77	98,34
	T4R2	67,16	99,08
	REO	56,15	94,49
	REC	34,49	90,27
T3R1 + T3R3	T1R2	62,45	98,25
	T2R2	54,07	97,13
	T3R2	64,85	99,45
	T4R2	59,96	98,80
	REO	44,59	92,84
	REC	29,17	89,17
T4R1 + T4R3	T1R2	65,41	98,89
	T2R2	73,84	98,71
	T3R2	58,39	96,59
	T4R2	70,11	98,89
	REO	46,97	93,39
	REC	30,83	87,52

Tabela 4.9 – Resultado para o protocolo P2: Acc-9 = Acurácia com 9 eletrodos; Acc-64 = Acurácia com 64 eletrodos. Modo de identificação.

eletrodos utilizados em (YANG; DERAVID; HOQUE, 2018), a maior acurácia obtida, de 93,82%, utilizou as mesmas tarefas para compor os dados de treino.

Novamente, a utilização dos 64 eletrodos disponíveis apresentou um melhor resultado que o trabalho de Yang, Deravi e Hoque (2018), o qual obteve uma acurácia de 99% no protocolo P3, enquanto a utilização dos mesmos 9 eletrodos apresentou um pior resultado.

4.5 *Padding Causal*

De maneira a replicar as convoluções causais dilatadas usadas no trabalho de Oord et al. (2016), o presente trabalho propôs utilizar *padding* causal em todas as camadas de convolução presentes no modelo neural.

A Tabela 4.11 contém os resultados de utilizar REO como tarefa de treino e REC como tarefa de teste, cenário este usado como base para a avaliação do modelo neural; e o penúltimo experimento do protocolo P3 presente em Yang, Deravi e Hoque (2018), já que este tinha

Tarefas de treino	Acc-9 (%)	Acc-64 (%)
T1R1	35,79	89,48
T1R3	41,51	92,16
T1R1 + T1R3	65,13	98,34
T2R1 + T2R2 + T2R3	79,43	97,42
T1R1 + T1R3 + T2R1	74,35	98,89
T1R1 + T2R1 + T2R2 + T2R3	84,96	99,54
T1R1 + T1R3 + T2R1 + T2R2	84,22	99,26
T1R1 + T1R3 + T2R1 + T2R2 + T2R3	84,87	99,63
T1R1 + T1R3 + T2R1 + T2R2 + T2R3 + T3R1	85,89	99,63
T1R1 + T1R3 + T2R1 + T2R2 + T2R3 + T3R1 + T4R1	93,82	99,72
T1R1 + T1R3 + T2R1 + T2R2 + T2R3 + T3R1 + T4R1 + T4R2	93,54	99,54

Tabela 4.10 – Resultado para o protocolo P3: Acc-9 = Acurácia com 9 eletrodos; Acc-64 = Acurácia com 64 eletrodos. Modo de identificação.

apresentado a maior acurácia obtida pelo modelo até então. Ambos os testes utilizaram todos os 64 eletrodos presentes na base de dados.

Tarefas de treino	Tarefa de teste	Acurácia (%)
REO	REC	96,15
T1R1 + T1R3 + T2R1 + T2R2 + T2R3 + T3R1 + T4R1	T1R2	99,81

Tabela 4.11 – Utilização de *padding* causal nas camadas de convolução.

Ao utilizar T1R1, T1R3, T2R1, T2R2, T2R3, T3R1 e T4R1 como tarefas de treino e T1R2 como tarefa de teste, uma acurácia de 99,81% foi obtida, sendo este o melhor resultado apresentado pelo modelo no trabalho proposto.

4.6 Comparação com os trabalhos relacionados

Nas Tabelas 4.12 e 4.13 se encontram comparações entre os resultados obtidos pelos trabalhos relacionados e pelo trabalho proposto, para o contexto de verificação e identificação biométrica, respectivamente.

Trabalho	Abordagem	# Indivíduos	EER (%)
Schons et al. (2017)	CNN	108	0,19
Mota et al. (2021)	CNN	109	0,10
Svetlakov et al. (2022)	CNN	109	14,63
Trabalho Proposto	CNN	109	0,07

Tabela 4.12 – Comparação entre trabalhos com foco modo de verificação.

Nelas, podemos observar que o trabalho proposto obteve o melhor resultado em comparação aos restantes, para ambos os modos biométricos, apresentado uma diferença aproximada

Trabalho	Abordagem	# Indivíduos	Acurácia (%)
Lai et al. (2019)	CNN	109	79,08
Fan, Shi e Li (2021)	CNN	109	99,32
Das, Maiorana e Campisi (2017)	CNN	40	98,80
Yang, Deravi e Hoque (2018)	Coefficientes de <i>wavelets</i>	109	99,00
Jijomon e Vinod (2021)	CNN + LSTM	20	99,53
Trabalho Proposto	CNN + LSTM	109	99,81

Tabela 4.13 – Comparação entre trabalhos com foco modo de identificação.

de 0,03% de EER e 0,3% de acurácia em relação aos trabalhos que obtiveram, respectivamente, o segundo melhor resultado para os modos de verificação e identificação.

4.7 Discussões

Nesta seção são apresentadas algumas questões referentes as conclusões obtidas ao observar os resultados dos experimentos presentes das Seções 4.1 (**Filtragem e Normalização**) a 4.5 (**Padding Causal**).

4.7.1 Performance de uma arquitetura CNN e LSTM/GRU e LSTM+CNN

Apesar de extração de características e classificação serem algumas das tarefas em que CNNs apresentam bons resultados (LIU, 2018; SCHONS et al., 2017), a utilização de uma CNN com arquitetura convencional (que possui camadas de convolução, *pooling*, normalização, densas e de *dropout*) não obteve bons resultados para o contexto biométrico de identificação, apresentando apenas 64,22% de acurácia. Da mesma forma, embora tarefas cujos dados variam ao longo do tempo sejam algumas das que RNNs baseadas na utilização de blocos de LSTM e GRU apresentam bons resultados (GRAVES; FERNÁNDEZ; SCHMIDHUBER, 2005), a utilização isolada destes blocos também não apresentou bons resultados para o contexto biométrico citado, resultando em apenas 57,25% e 67,34% de acurácia, respectivamente.

Entretanto, o fusionamento das arquiteturas CNN e LSTM chegou a apresentar um resultado de 99,72% de acurácia. Isso pode ser devido a natureza de um sinal de EEG, que varia com o tempo, aliado à necessidade de extração de características para distinguir indivíduos em um sistema biométrico.

4.7.2 Performance por região cerebral

Ao examinar a Tabela 4.8, podemos inferir que a utilização de eletrodos localizados na região occipital do cérebro produz um desempenho superior em comparação com outras regiões cerebrais. Essa observação sugere que a região occipital do cérebro pode ter uma maior capacidade de distinguir indivíduos. Além disso, se uma quantidade maior de eletrodos presentes

na região occipital for selecionada, pode ser possível obter um resultado similar ao que utiliza todos os 64 eletrodos, porém, com um menor custo computacional.

4.7.3 Composição das tarefas de treino

Os resultados da Tabela 4.9 validam algumas das conclusões obtidas em (YANG; DERAVI; HOQUE, 2018). Em particular, a inclusão de diferentes tarefas na composição dos dados de treinamento não necessariamente leva a uma piora dos resultados obtidos pela rede neural. No entanto, estes resultados podem ser significativamente piorados se as tarefas selecionadas para compor os dados de treinamento e validação forem tarefas motoras ou imaginárias, enquanto as tarefas selecionadas para compor os dados de teste forem REO ou REC. Isso pode sugerir que, quando uma pessoa está em repouso, ela não está focada em realizar alguma tarefa específica, e seus sinais de EEG não possuem tanto valor para diferenciá-la de outra.

Já os resultados apresentados pela Tabela 4.10 confirmam que, em geral, adicionar mais tarefas na composição dos dados de treinamento tende a aumentar a acurácia obtida pela rede. No entanto, foi observada uma piora nos resultados no último cenário do protocolo P3 (adição de T4R2 nos dados de treino e validação), tanto em (YANG; DERAVI; HOQUE, 2018) quanto no trabalho proposto, o que sugere que pode haver alguma relação entre as tarefas selecionadas para treinamento e teste e que é importante selecionar as tarefas de treinamento com cuidado para evitar tal piora.

4.7.4 Utilização de *padding* causal

A utilização de *padding* causal nas camadas de convolução do modelo melhorou os resultados obtidos, e uma acurácia de 99,81% de foi atingida. Então, considerando os sinais de EEG como uma série temporal, pode ser benéfico para o modelo manter uma relação de causalidade ao utilizar apenas parte dos dados de entrada anterior a um momento t para produzir dados de saída equivalentes a esse mesmo momento t .

5 Considerações Finais

Este capítulo descreve de forma sucinta os resultados obtidos e sua relação com os objetivos estabelecidos na Seção 1.2. Também são apresentadas propostas de continuidade para trabalhos futuros.

5.1 Conclusão

Neste trabalho, foram exploradas as performances de uma CNN, de blocos de LSTM e de GRU contíguos, e do fusionamento entre estes tipos de arquitetura, em um sistema biométrico baseado em sinais de EEG, com foco no modo de identificação. Ao utilizar *padding* causal nas camadas de convolução do modelo neural, um resultado estado da arte de 99,81% de acurácia foi obtido.

Os resultados apresentados demonstram que há uma melhora da acurácia obtida pelo modelo após a aplicação de um filtro FIR passa-banda de 30-50Hz. Deste modo, os sinais de EEG que pertencem às frequências beta e gama, dentro da faixa especificada, podem ter um maior valor para diferenciar indivíduos em um sistema biométrico, se comparado com as demais frequências delta, teta e alfa.

Ademais, através dos protocolos P1, P2 e P3 propostos por [Yang, Deravi e Hoque \(2018\)](#), foi possível observar como a composição das tarefas de treino e teste afetam a acurácia do modelo. Ao conseguir resultados superiores ao de [Yang, Deravi e Hoque \(2018\)](#) nos três protocolos, o modelo proposto também se provou robusto ao utilizar sinais de EEG provenientes de diferentes tarefas.

Finalmente, a metodologia aplicada no trabalho proposto, que combina técnicas de pré-processamento dos sinais de EEG, fusionamento de CNN com blocos de LSTM e *padding* causal, provou ser apta para a construção de sistemas biométricos no modo de identificação.

5.2 Trabalhos Futuros

Na perspectiva de trabalhos futuros, ainda há espaço para diminuir o tamanho de janela necessário para o treinamento do modelo, visto que 12 segundos é inviável em sistemas biométricos do dia-a-dia, os quais iriam fazer os usuários esperarem por uma grande quantidade de tempo. Deste modo, mudanças na arquitetura da rede neural proposta, como no formato de entrada e nos parâmetros de cada uma das camadas, ainda podem ser exploradas.

5.3 Publicações Realizadas

É válido acrescentar que o seguinte trabalho, originado da metodologia proposta, foi aceito para apresentação em uma conferência nacional:

- Freitas et al. Rede Neural Convolucional e LSTM para Biometria Baseada em EEG no Modo de Identificação. XXII SIMPÓSIO BRASILEIRO DE COMPUTAÇÃO APLICADA À SAÚDE, 2022.

Referências

- ABHANG, P. A.; GAWALI, B. W.; MEHROTRA, S. C. Chapter 2 - technological basics of eeg recording and operation of apparatus. In: ABHANG, P. A.; GAWALI, B. W.; MEHROTRA, S. C. (Ed.). Introduction to EEG- and Speech-Based Emotion Recognition. Academic Press, 2016. p. 19–50. ISBN 978-0-12-804490-2. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780128044902000026>>.
- AGRAWAL, N.; KUMAR, A.; BAJAJ, V.; SINGH, G. K. Design of digital iir filter: A research survey. Applied Acoustics, Elsevier, v. 172, p. 107669, 2021.
- ALBAWI, S.; MOHAMMED, T. A.; AL-ZAWI, S. Understanding of a convolutional neural network. In: IEEE. 2017 international conference on engineering and technology (ICET). [S.l.], 2017. p. 1–6.
- BENGIO, Y. Practical recommendations for gradient-based training of deep architectures. CoRR, abs/1206.5533, p. 437–478, 2012. Disponível em: <<http://arxiv.org/abs/1206.5533>>.
- BOBKOWSKA, K.; NAGATY, K.; PRZYBORSKI, M. Incorporating iris, fingerprint and face biometric for fraud prevention in e-passports using fuzzy vault. IET Image Processing, IET, v. 13, n. 13, p. 2516–2528, 2019.
- CARRIÓN-OJEDA, D.; FONSECA-DELGADO, R.; PINEDA, I. Analysis of factors that influence the performance of biometric systems based on eeg signals. Expert Systems with Applications, Elsevier, v. 165, p. 113967, 2021.
- CHANDRA, A.; CHATTOPADHYAY, S. Design of hardware efficient fir filter: A review of the state-of-the-art approaches. Engineering Science and Technology, an International Journal, Elsevier, v. 19, n. 1, p. 212–226, 2016.
- DAMAŠEVIČIUS, R.; MASKELIŪNAS, R.; KAZANAVIČIUS, E.; WOŹNIAK, M. Combining cryptography with EEG biometrics. Computational Intelligence and Neuroscience, Hindawi, v. 2018, p. 1867548:1–1867548:11, 2018. Disponível em: <<https://doi.org/10.1155/2018/1867548>>.
- DAS, R.; MAIORANA, E.; CAMPISI, P. Eeg biometrics using visual stimuli: A longitudinal study. IEEE Signal Processing Letters, IEEE, v. 23, n. 3, p. 341–345, 2016.
- DAS, R.; MAIORANA, E.; CAMPISI, P. Visually evoked potential for EEG biometrics using convolutional neural network. In: 25th European Signal Processing Conference, EUSIPCO 2017, August 28 - September 2, 2017. Kos, Greece: IEEE, 2017. p. 951–955. Disponível em: <<https://doi.org/10.23919/EUSIPCO.2017.8081348>>.
- DENG, J.; DONG, W.; SOCHER, R.; LI, L.-J.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. [S.l.: s.n.], 2009. p. 248–255.
- ELIAS, I.; RUBIO, J. d. J.; CRUZ, D. R.; OCHOA, G.; NOVOA, J. F.; MARTINEZ, D. I.; MUÑIZ, S.; BALCAZAR, R.; GARCIA, E.; JUAREZ, C. F. Hessian with mini-batches for electrical demand prediction. Applied Sciences, MDPI, v. 10, n. 6, p. 2036, 2020.

- FAN, Y.; SHI, X.; LI, Q. Cnn-based personal identification system using resting state electroencephalography. *Computational Intelligence and Neuroscience*, Hindawi, London, GBR, v. 2021, jan 2021. ISSN 1687-5265. Disponível em: <<https://doi.org/10.1155/2021/1160454>>.
- FRASCHINI, M.; HILLEBRAND, A.; DEMURU, M.; DIDACI, L.; MARCIALIS, G. L. An eeg-based biometric system using eigenvector centrality in resting state brain networks. *IEEE Signal Processing Letters*, IEEE, v. 22, n. 6, p. 666–670, 2014.
- GOLDBERGER, A. L.; AMARAL, L. A.; GLASS, L.; HAUSDORFF, J. M.; IVANOV, P. C.; MARK, R. G.; MIETUS, J. E.; MOODY, G. B.; PENG, C.-K.; STANLEY, H. E. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, Am Heart Assoc, v. 101, n. 23, p. e215–e220, 2000.
- GRAVES, A.; FERNÁNDEZ, S.; SCHMIDHUBER, J. Bidirectional LSTM networks for improved phoneme classification and recognition. In: *15th International Conference on Artificial Neural Networks*. [S.l.: s.n.], 2005. p. 799–804.
- HAWKING, S. *Time*. 2010. Acessado em 02 de Março de 2023. Disponível em: <<https://time.com/5198842/stephen-hawking-quotes-universe-life/>>.
- HE, K.; ZHANG, X.; REN, S.; SUN, J. *Deep Residual Learning for Image Recognition*. arXiv, 2015. Disponível em: <<https://arxiv.org/abs/1512.03385>>.
- HU, J.; SHEN, L.; SUN, G. Squeeze-and-excitation networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2018.
- IDE, H.; KURITA, T. Improvement of learning for cnn with relu activation by sparse regularization. In: *IEEE. 2017 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2017. p. 2684–2691.
- ISMAIL, W. W.; HANIF, M.; MOHAMED, S.; HAMZAH, N.; RIZMAN, Z. I. Human emotion detection via brain waves study by using electroencephalogram (eeg). *International Journal on Advanced Science, Engineering and Information Technology*, v. 6, n. 6, p. 1005–1011, 2016.
- JAIN, A. K.; BOLLE, R.; PANKANTI, S. *Biometrics: personal identification in networked society*. [S.l.]: Springer Science & Business Media, 2006. v. 479.
- JAMIL, Z.; JAMIL, A.; MAJID, M. Artifact removal from eeg signals recorded in non-restricted environment. *Biocybernetics and Biomedical Engineering*, Elsevier, v. 41, n. 2, p. 503–515, 2021.
- JAVAID, A.; NIYAZ, Q.; SUN, W.; ALAM, M. A deep learning approach for network intrusion detection system. In: *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*. [S.l.: s.n.], 2016. p. 21–26.
- JIJOMON, C.; VINOD, A. Person-identification using familiar-name auditory evoked potentials from frontal eeg electrodes. *Biomedical Signal Processing and Control*, Elsevier, v. 68, p. 102739, 2021.
- KETKAR, N. Stochastic gradient descent. In: *Deep learning with Python*. [S.l.]: Springer, 2017. p. 113–132.

- LAI, C. Q.; IBRAHIM, H.; ABDULLAH, M. Z.; ABDULLAH, J. M.; SUANDI, S. A.; AZMAN, A. Arrangements of resting state electroencephalography as the input to convolutional neural network for biometric identification. Computational Intelligence and Neuroscience, Hindawi, v. 2019, p. 7895924:1–7895924:10, 2019. Disponível em: <<https://doi.org/10.1155/2019/7895924>>.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. nature, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015.
- LECUN, Y. A.; BOTTOU, L.; ORR, G. B.; MÜLLER, K.-R. Efficient backprop. In: Neural networks: Tricks of the trade. [S.l.]: Springer, 2012. p. 9–48.
- LEE, Y.-Y.; HSIEH, S. Classifying different emotional states by means of eeg-based functional connectivity patterns. PloS one, Public Library of Science San Francisco, USA, v. 9, n. 4, p. e95415, 2014.
- LIN, H. W.; TEGMARK, M. Critical behavior in physics and probabilistic formal languages. Entropy, Multidisciplinary Digital Publishing Institute, v. 19, n. 7, p. 299, 2017.
- LIU, Y. H. Feature extraction and image recognition with convolutional neural networks. Journal of Physics: Conference Series, v. 1087, n. 6, p. 062032, 09 2018.
- LUMINI, A.; NANNI, L. Overview of the combination of biometric matchers. Information Fusion, Elsevier, v. 33, p. 71–85, 2017.
- MA, L.; MINETT, J. W.; BLU, T.; WANG, W. S. Resting state eeg-based biometrics for individual identification using convolutional neural networks. In: IEEE. 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). [S.l.], 2015. p. 2848–2851.
- MAIORANA, E. Deep learning for eeg-based biometric recognition. Neurocomputing, Elsevier, v. 410, p. 374–386, 2020.
- MOTA, M. R.; SILVA, P. H.; LUZ, E. J.; MOREIRA, G. J.; SCHONS, T.; MORAES, L. A.; MENOTTI, D. A deep descriptor for cross-tasking eeg-based recognition. PeerJ Computer Science, PeerJ Inc., v. 7, p. e549, 2021.
- OBAIDAT, M. S.; RANA, S. P.; MAITRA, T.; GIRI, D.; DUTTA, S. Biometric security and internet of things (iot). In: Biometric-based physical and cybersecurity systems. [S.l.]: Springer, 2019. p. 477–509.
- OORD, A. V. D.; DIELEMAN, S.; ZEN, H.; SIMONYAN, K.; VINYALS, O.; GRAVES, A.; KALCHBRENNER, N.; SENIOR, A. W.; KAVUKCUOGLU, K. Wavenet: A generative model for raw audio. SSW, v. 125, p. 2, 2016.
- ROJAS, R. The backpropagation algorithm. In: Neural networks. [S.l.]: Springer, 1996. p. 149–182.
- SARIN, S.; MITTAL, A.; CHUGH, A.; SRIVASTAVA, S. Cnn-based multimodal touchless biometric recognition system using gait and speech. Journal of Intelligent & Fuzzy Systems, IOS Press, v. 42, n. Preprint, p. 1–10, 03 2021.

- SCHALK, G.; MCFARLAND, D. J.; HINTERBERGER, T.; BIRBAUMER, N.; WOLPAW, J. R. Bci2000: a general-purpose brain-computer interface (bci) system. IEEE Transactions on biomedical engineering, IEEE, v. 51, n. 6, p. 1034–1043, 2004.
- SCHONS, T.; MOREIRA, G. J. P.; SILVA, P. H. L.; COELHO, V. N.; LUZ, E. J. da S. Convolutional network for eeg-based biometric. In: 22nd Iberoamerican Congress on Pattern Recognition. [S.l.: s.n.], 2017. p. 601–608.
- SHEPHARD, R. W.; FÄRE, R. The law of diminishing returns. In: Production theory. [S.l.]: Springer, 1974. p. 287–318.
- SHERSTINSKY, A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. Physica D: Nonlinear Phenomena, Elsevier, v. 404, p. 132306, 2020.
- SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, JMLR. org, v. 15, n. 1, p. 1929–1958, 2014.
- SUN, M.; SONG, Z.; JIANG, X.; PAN, J.; PANG, Y. Learning pooling for convolutional neural network. Neurocomputing, Elsevier, v. 224, p. 96–104, 2017.
- SUN, Y.; LO, F. P.-W.; LO, B. Eeg-based user identification system using 1d-convolutional long short-term memory neural networks. Expert Systems with Applications, Elsevier, v. 125, p. 259–267, 2019.
- SUTSKEVER, I.; MARTENS, J.; DAHL, G.; HINTON, G. On the importance of initialization and momentum in deep learning. In: DASGUPTA, S.; MCALLESTER, D. (Ed.). Proceedings of the 30th International Conference on Machine Learning. Atlanta, Georgia, USA: PMLR, 2013. (Proceedings of Machine Learning Research, 3), p. 1139–1147.
- SVETLAKOV, M.; KOVALEV, I.; KONEV, A.; KOSTYUCHENKO, E.; MITSEL, A. Representation learning for eeg-based biometrics using hilbert–huang transform. Computers, MDPI, v. 11, n. 3, p. 47, 2022.
- WU, Y.; QIN, Y. Machine translation of english speech: Comparison of multiple algorithms. Journal of Intelligent Systems, De Gruyter, v. 31, n. 1, p. 159–167, 2022.
- XU, Y.; GOODACRE, R. On splitting training and validation set: A comparative study of cross-validation, bootstrap and systematic sampling for estimating the generalization performance of supervised learning. Journal of analysis and testing, Springer, v. 2, n. 3, p. 249–262, 2018.
- YANG, S.; DERAVIDI, F.; HOQUE, S. Task sensitivity in eeg biometric recognition. Pattern Analysis and Applications, Springer, v. 21, n. 1, p. 105–117, 2018.
- YU, Y.; SI, X.; HU, C.; ZHANG, J. A review of recurrent neural networks: Lstm cells and network architectures. Neural computation, MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info . . . , v. 31, n. 7, p. 1235–1270, 2019.