

# Reconhecimento de padrões biométricos utilizando máquinas de aprendizado profundo

Gabriel Anísio S. Soares<sup>1</sup>, Bruno Otávio P. Prado<sup>1</sup>, Gilton José Ferreira da Silva<sup>1</sup>,  
Hendrik Teixeira Macedo<sup>1</sup>, Leonardo Nogueira Matos<sup>1</sup>

<sup>1</sup>Departamento de Computação  
Universidade Federal de Sergipe (UFS) – São Cristóvão, SE – Brasil

{gabriel.soares, bruno, gilton, hendrik, leonardo}@dcomp.ufs.br

**Abstract.** *The brain-computer interface is one of the emerging fields of human-computer interaction due to its broad spectrum of applications, especially those dealing with human cognition. In this work, electroencephalography (EEG) is used as a base data to classify eye condition based on Long Short Term Memory (LSTM) networks. For purposes of benchmarking, the EEG eye state data set available in the UCI machine learning repository was used. The results indicated that the model is applicable to the data classification and that its performance is good compared to the computationally more expensive models.*

**Resumo.** *A interface cérebro-computador é um dos campos emergentes da interação homem-computador devido ao seu amplo espectro de aplicações, especialmente as que lidam com a cognição humana. Neste trabalho, a eletroencefalografia (EEG) é usada como dado base para classificar o estado dos olhos baseado em redes Long Short Term Memory (LSTM). Para fins de benchmarking, foi utilizado o conjunto de dados do estado do olho do EEG disponível no repositório de aprendizado de máquina da UCI. Os resultados obtidos indicaram que o modelo é aplicável para a classificação dos dados e que seu desempenho é bom comparado aos modelos mais caros computacionalmente.*

## 1. Introdução

A interface cérebro-computador é um dos campos emergentes da interação homem-computador. Tem um amplo espectro de aplicações, sendo as mais proeminentes as aplicações industriais e médicas [Narejo et al. 2016]. Muitos estudos sobre sinais de eletroencefalografia (EEG), que é o registro da atividade elétrica espontânea do cérebro durante um período de tempo, trouxeram descobertas importantes e úteis para a classificação do estado cognitivo humano.

De acordo com [Wang et al. 2014], a classificação do estado ocular do EEG tem sido aplicada com sucesso nas áreas de identificação do estado de sonambulismo, detecção de sonolência na condução de veículo, detecção de crises epiléticas e déficit de atenção pacientes com transtorno de hiperatividade (TDAH), identificação de características de estresse, detecção de piscar de olho humano, etc. Essas aplicações indicam a importância da pesquisa na análise do sinal do EEG no estado dos olhos.

Os dados que descrevem o estado ocular do EEG pertencem ao tipo contínuo de dados de séries temporais. Existem várias técnicas de aprendizado de máquina e abordagens estatísticas que podem ser empregadas para resolver os problemas de classificação

com esses dados de séries temporais. Este trabalho tem como objetivo propor mais uma abordagem para a identificação do estado do olho do EEG usando redes neurais recorrentes com memória de curto prazo (LSTM).

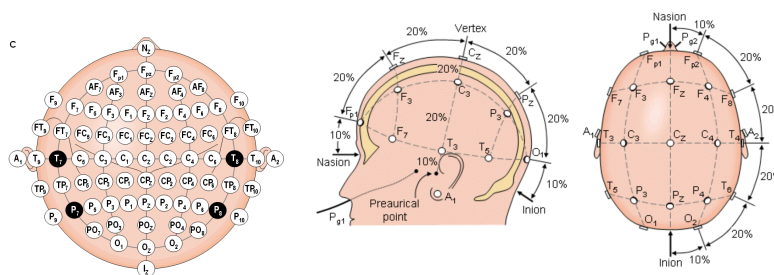
Para facilitar a identificação dos pontos fortes e fracos da metodologia de aprendizado de máquina utilizada neste trabalho, será utilizado o conjunto de dados de *benchmark EEG Eye State Data Set*<sup>1</sup> disponível publicamente [Rösler and Suendermann 2013].

## 2. Revisão da Literatura

### 2.1. Eletroencefalografia (EEG)

Eletroencefalografia (EEG) é a medida da atividade elétrica produzida pelo cérebro, registrada por eletrodos colocados no couro cabeludo. Esta atividade é registrada, condicionada e exibida na tela do computador. O EEG é utilizado para diagnosticar lesões cerebrais, tumores e outras anomalias de forma não invasiva, além de ser utilizado em pesquisas de neurociência, ciência cognitiva, psicologia cognitiva, neurolinguística e pesquisa psicofisiológica [Northrop 2012].

As localizações topográficas dos eletrodos são padronizadas e geralmente aderem ao sistema internacional 10-20 [Acharya et al. 2016], vide Figura 1. O “10” e o “20” referem-se ao fato de que as distâncias reais entre os eletrodos adjacentes são 10% ou 20% da distância total da frente ou da direita e esquerda do crânio. Cada posição de gravação é identificada por uma combinação de letras e números. Por exemplo, F = Frontal, Fp = Pós frontal, C = Central, P = Parietal, O = Occipital, T = Temporal, A = Lóbulo da orelha, Pg = Faríngea, Z = Zero (linha média). Os números ímpares estão no lado esquerdo da cabeça, e os números pares estão à direita.

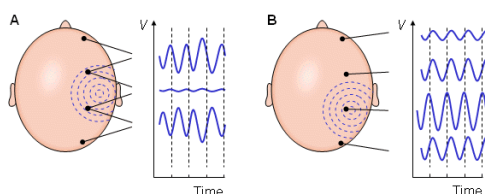


**Figura 1. (Esquerda) Localização e nomenclatura dos eletrodos, de acordo com o padrão da *American Electroencephalographic Society*. (Centro e Direita) Vista lateral esquerda e de topo do posicionamento dos eletrodos de acordo com o padrão. Fonte: [Malmivuo and Plonsey 1995]**

Segundo [Northrop 2012], por ser geralmente registrada a partir do couro cabeludo, a atividade elétrica do córtex cerebral deve passar pelas membranas pia e dura-máter, líquido cefalorraquidiano, crânio e couro cabeludo. A atenuação considerável e a média espacial ocorrem por causa dessas estruturas anatômicas em relação à atividade elétrica. Os maiores potenciais de EEG registrados no couro cabeludo são de aproximadamente 150  $\mu$ V-pico.

<sup>1</sup>Disponível em <http://archive.ics.uci.edu/ml/datasets/EEG+Eye+State>.

Na aferição EEG podem ser usadas configurações bipolares ou unipolares de eletrodos. Na primeira configuração (ver Figura 2, A), a diferença de potencial entre um par de eletrodos é medida. Na segunda (ver Figura 2, B), o potencial de cada eletrodo é comparado a um eletrodo neutro ou à média de todos os eletrodos (média espacial).



**Figura 2. (A) Configuração bipolar e (B) Configuração unipolar. Fonte: [Malmivuo and Plonsey 1995]**

Os ritmos cerebrais estão, tradicionalmente, agrupados em bandas de frequência, que incluem [Cohen 2014, Northrop 2012]:

- as ondas delta ( $\delta$ ) são definidas para frequências de 2 a 4 Hz, ocorrem em adultos em sono profundo;
- as ondas teta ( $\theta$ ) são definidas para frequências de 4 a 8 Hz, ocorrem durante o sono em adultos e em crianças pré-púberes, também pode ser vista durante a meditação;
- as ondas alfa ( $\alpha$ ) são definidas para frequências de 8 a 14 Hz, ocorrem em adultos conscientes, mas relaxados, com os olhos fechados, desaparece quando os olhos estão abertos e o indivíduo se concentra em alguma tarefa;
- as ondas beta ( $\beta$ ) são definidas para frequências de 14 a 30 Hz. As ondas beta são subdivididas nos tipos I e II: O beta do tipo I desaparece durante a atividade mental intensa, enquanto as ondas beta tipo II aparecem;

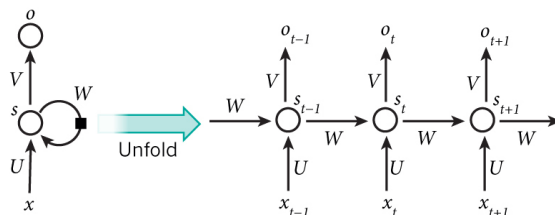
Vale ressaltar que não existem limites precisos que definem as bandas de frequência. Além disso, as diferenças individuais nas frequências de pico estão ligadas a uma série de características individuais, incluindo estrutura do cérebro, idade, capacidade de memória de trabalho e química do cérebro [Cohen 2014].

Diferentes padrões de EEG em áreas distintas do cérebro estão intimamente associados com o nível de consciência da pessoa. À medida que a atividade aumenta, o EEG muda para maior frequência dominante e menor amplitude. Estados de excitação, falta de atenção, sonolência, diferentes estágios do sono, presença de diferentes tipos de drogas e toda uma série de estados clínicos se manifestam distintamente no EEG. Quando os olhos estão fechados, as ondas alfa tendem a dominar o EEG. Quando a pessoa adormece, a frequência de EEG dominante diminui. No sono profundo, o EEG possui amplitudes grandes e frequência baixa (ondas delta) [Malmivuo and Plonsey 1995]. Durante a meditação, geralmente ocorrem aumentos de amplitude nas ondas teta e alfa e diminuição da frequência geral [Cahn and Polich 2006]. Nenhuma atividade cerebral pode ser detectada por um paciente com morte cerebral completa.

## **2.2. Redes Neurais Recorrentes**

Uma rede neural recorrente é uma rede neural artificial em que as conexões entre as unidades ocultas formam ciclos. Redes neurais recorrentes ou RNNs são um tipo específico de rede neural projetada para problemas que envolvem sequências [Graves 2012].

Dada uma rede *perceptron feedforward*, uma rede neural recorrente pode ser pensada como a adição de *loops* à arquitetura, vide Figura 3. Desse modo, a entrada da RNN inclui também a entrada da etapa anterior. A rede processa a entrada atual, e usa um *loop* de *feedback* para levar em consideração as entradas do passo anterior, também chamado de passado recente, para o contexto. As conexões recorrentes adicionam estado, ou memória, à rede e permitem que ela lide com abstrações mais amplas das sequências de entrada.



**Figura 3.** Uma rede neural recorrente e seu desdobramento no tempo exibindo os componentes dos cálculos envolvidos na propagação *forward* [LeCun et al. 2015].

As equações de propagação *forward* para a RNN exposta na Figura 3 são definidas como segue:

$$\begin{aligned} \mathbf{s}_t &= \phi_s(\mathbf{U}\mathbf{x}_t + \mathbf{W}\mathbf{o}_{t-1} + b_s) \\ \hat{\mathbf{o}}_t &= \phi_o(\mathbf{V}\mathbf{s}_t + b_o) \end{aligned} \quad (1)$$

Onde  $\mathbf{x}_t$ ,  $\mathbf{s}_t$  e  $\mathbf{o}_t$  são *arrays* das camadas de entrada, oculta e saída, respectivamente;  $\hat{\mathbf{o}}_t$  é o *array* da camada de saída prevista.  $\mathbf{W}$ ,  $\mathbf{U}$  e  $\mathbf{V}$  são matrizes de pesos (conexões entre neurônios);  $\phi_s$  e  $\phi_o$  são as funções de ativação da camada oculta e da camada de saída, respectivamente; e  $b_s$  e  $b_o$  são o *bias* para a camada oculta e de saída, respectivamente. Os subíndices  $t$  referem-se ao período de tempo.

Ao lidar com abstrações mais amplas das sequências de entrada, as RNNs *vanilla* podem ficar suscetíveis a dificuldades de treinamento para resolver problemas que exigem o aprendizado de dependências temporais longas. Isso ocorre porque o gradiente da função de perda decai exponencialmente com o tempo (chamado de problema do desaparecimento do gradiente) [Graves 2012]. Para tratar esse problema utiliza-se um conjunto de "portas" para controlar o fluxo de informações entre as unidades.

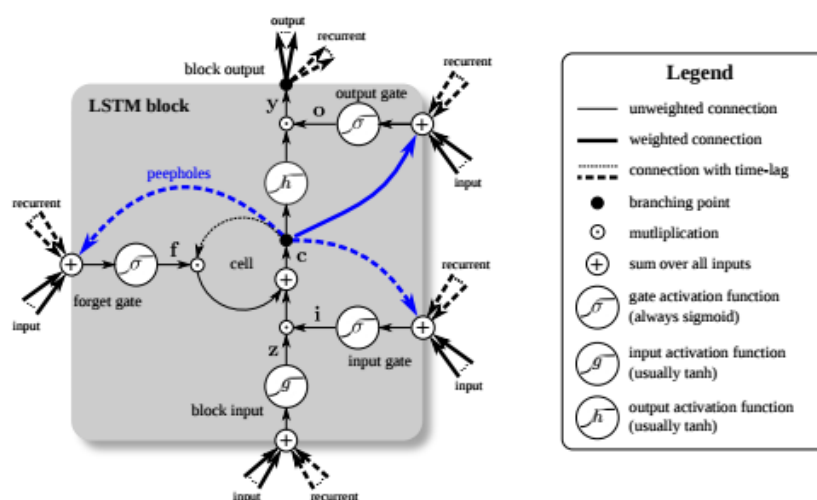
### 2.3. Redes *Long Short Term Memory* (LSTM)

As redes LSTM são um tipo especial de RNN, capaz de aprender dependências de longo prazo. De acordo com [Greff et al. 2017], uma LSTM *vanilla* possui três portões (entrada, esquecimento, saída), entrada de bloco, uma única célula, uma função de ativação de saída e conexões do olho mágico. Um esquema do bloco LSTM *vanilla* pode ser visto na Figura 4. Estes componentes são descritos a seguir [Fonseca 2016]:

- Portão de entrada (*input gate*) – pondera a importância de cada característica do vetor de entrada no tempo  $t$ ,  $\mathbf{x}_t$  e o vetor de saída no passo anterior  $\mathbf{y}_{t-1}$ , produzindo uma saída;

- Portão de entrada do bloco (*block input gate*) – controla o fluxo de informações do portão de entrada para a célula de memória (também recebe o vetor de entrada e o vetor de saída anterior como entradas), sua dinâmica é controlada por um conjunto diferente de pesos;
- Portão de esquecimento (*forget gate*) – controla o conteúdo da célula de memória, seja para configurá-los ou redefini-los;
- Portão de saída do bloco (*block output gate*) – controla o fluxo de informações do neurônio LSTM;
- Célula de memória (*cell memory*) – é o elemento de memória do neurônio, onde o estado anterior é mantido e atualizado de acordo com a dinâmica dos portões que se conectam a ele. Também é de onde as conexões *peepholes* se originam;
- Ativação de saída (*output activation*) – a saída da célula de memória passa por esta função de ativação;
- Olho mágico (*peepholes*) – ligações diretas a partir da célula de memória que permitem aos portões ‘espiar’ os estados da célula de memória.

A saída do bloco é conectada de forma recorrente de volta à entrada do bloco e a todos os portões.



**Figura 4. Esquema detalhado da unidade de um bloco de Memória de Longo Prazo como usado nas camadas ocultas de uma rede neural recorrente. Fonte: [Greff et al. 2017].**

Os LSTMs abordam o problema do desaparecimento do gradiente, comumente encontrado nas RNNs, incorporando funções de ativação em sua dinâmica de estado. Em cada passo de tempo, um LSTM mantém um vetor oculto  $s$  e um vetor de memória  $c$  responsável pelo controle de atualizações e saídas de estados [Zaremba et al. 2014].

Formalmente, a computação no tempo  $t$  é definida como segue [Kalchbrenner et al. 2015]:

$$\begin{aligned}
\mathbf{i} &= \sigma(\mathbf{U}^i * x_t + \mathbf{s}_{t-1}W^i) \\
\mathbf{f} &= \sigma(\mathbf{U}^f * x_t + \mathbf{s}_{t-1}W^f) \\
\mathbf{o} &= \sigma(\mathbf{U}^o * x_t + \mathbf{s}_{t-1}W^o) \\
\mathbf{g} &= \tanh(\mathbf{U}^g * x_t + \mathbf{s}_{t-1}W^g) \\
\mathbf{c}_t &= \mathbf{c}_{t-1} \odot \mathbf{f} + \mathbf{g} \odot \mathbf{i} \\
\mathbf{s}_t &= \tanh(\mathbf{c}_t) \odot \mathbf{o}
\end{aligned} \tag{2}$$

Onde  $\mathbf{i}$ ,  $\mathbf{f}$  e  $\mathbf{o}$  são chamados de portas de entrada, esquecimento e saída, respectivamente;  $\mathbf{g}$  é um estado oculto "candidato" que é computado com base na entrada atual e no estado oculto anterior;  $\mathbf{c}_t$  é a memória interna da unidade, uma combinação da memória anterior  $\mathbf{c}_{t-1}$  multiplicada pelo portão de esquecimento, e o estado oculto recentemente calculado  $\mathbf{g}$ , multiplicado pelo portão de entrada;  $\mathbf{s}_t$  é o estado oculto de saída obtido pela multiplicação da memória com o portão de saída;  $\sigma$  é a função logística sigmoide;  $\mathbf{W}^i, \mathbf{W}^f, \mathbf{W}^o, \mathbf{W}^g$  são matrizes de pesos recorrentes e  $\mathbf{U}^i, \mathbf{U}^f, \mathbf{U}^o, \mathbf{U}^g$  são matrizes de projeção.

### 3. Metodologia

A metodologia empregada neste projeto contempla duas fases: a primeira envolve a classificação de padrões no EEG utilizando redes profundas LSTM, e a segunda envolve a comparação dos resultados obtidos com os da literatura.

Na primeira etapa serão implementados modelos de redes profundas LSTM em linguagem Python, utilizando a API de redes neurais de alto nível Keras [Chollet et al. 2015] com o *backend* do *framework* TensorFlow [Abadi et al. 2015]. Os testes iniciais serão realizados com bases de dados disponíveis para download gratuitamente, como a base EEG disponível no repositório UCI<sup>2</sup>, intitulada *EEG Eye State Data Set*, na qual todos os dados são de uma medição de EEG contínua com o dispositivo *EEI Neuroheadset Emotiv*<sup>3</sup>. Foram utilizados 14 eletrodos nas posições AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4. Como duração da medição de 117 segundos e o estado do olho foi detectado através de uma câmera durante a medição do EEG e adicionado manualmente ao arquivo depois de analisar os quadros de vídeo. '1' indica o olho fechado e '0' o estado do olho aberto. Todos os valores estão em ordem cronológica com o primeiro valor medido na parte superior dos dados.

Durante os experimentos, a base foi particionada em conjuntos de treinamento, validação e teste com as respectivas divisões de 60%, 20% e 20%, com ordem aleatória de seleção, além disso, também foi utilizada a técnica de validação de modelo *k-fold cross-validation*, com  $k = 10$ , para avaliar como os resultados serão generalizados para um conjunto de dados independente.

Foram utilizados dois modelos nos experimentos, o primeiro utilizando LSTM e o segundo *Bidirectional Long Short Term Memory* (BLSTM), que é uma variante da rede LSTM com aprendizado de dependências de longo prazo bidirecionais ("passado" e

<sup>2</sup>Disponível em <http://archive.ics.uci.edu/ml/datasets/EEG+Eye+State>.

<sup>3</sup>Mais informações em <https://www.emotiv.com/epoc/>.

”futuro”) entre as etapas de tempo (úteis quando se deseja que a rede aprenda com a série temporal completa em cada etapa de tempo).

Por fim, os resultados do modelo proposto são comparados com os de trabalhos relacionados que utilizaram a mesma base de dados.

#### 4. Resultados

Ao efetuar uma análise exploratória da base de dados notou-se a presença de valores fora do normal, foi realizado o tratamento da base, removendo estes *outliers*, a base resultante tratada tem 8172 instâncias válidas de ”olho aberto”(’0’) e 6720 instâncias válidas de ”olho fechado”(’1’).

Em seguida verificou-se os coeficientes de correlação (matriz de correlação), ver Figura 5, para tentar identificar as *features* (colunas do conjunto de dados) que têm uma alta correlação com a variável de *target*<sup>4</sup>. No caso desta base de dados, as correlações se mantêm próximas a zero, ou seja, provavelmente não se encontrará uma base ótima para representar os dados de forma compacta, isso pode ser observado verificando as cores frias na última linha (e na última coluna, já que se trata de uma matriz simétrica), tendo em vista que a classe ou variável de decisão (*target*) está associada à última posição nesta matriz.

Também quantificou-se a relação de dependência linear (matriz de dispersão) entre as *features*, que fornece um resumo gráfico das relações no conjunto de dados, verificou-se que, os histogramas das variáveis parecem estar normalmente distribuídas. Além disso, as relações entre alguns pares sugerem um padrão linear (pontos mais quentes no gráfico de calor da matriz de correlação).

Antes de utilizar os dados nos modelos, em uma etapa complementar à remoção de *outliers*<sup>5</sup> do conjunto de dados, foi realizado um escalonamento das instâncias para que permanecessem entre um valor mínimo e máximo, neste caso entre zero e um. Isto porquê, em geral, os algoritmos de aprendizado se beneficiam da padronização do conjunto de dados.

O primeiro modelo era formado por uma camada LSTM com 100 unidades, que dispensava aleatoriamente, uma taxa de 50% de unidades de entrada, para 0 a cada atualização durante o treinamento (previne *overfitting*) antes da camada de saída. O modelo avaliado obteve acurácia de  $92.57\% \pm 0.72\%$  (*10-fold*). As demais métricas são apresentadas na Tabela 1.

O segundo modelo era formado por uma camada BLSTM, também com 100 uni-

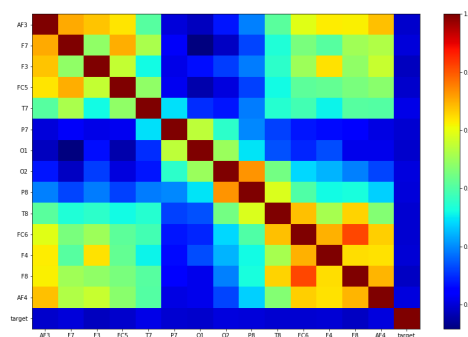


Figura 5. Matriz de correlação da base de dados representada como um mapa de calor.

<sup>4</sup>Variável de *target*, no contexto deste trabalho, é a variável que é ou deve ser a saída. Por exemplo, pode ser 0 ou 1 binário (tarefa de classificação) ou pode ser uma variável contínua (tarefa de regressão).

<sup>5</sup>*Outliers* são valores extremos muito além das outras observações do conjunto de dados.

**Tabela 1. Resumo da precisão, recall, F1-score para cada classe do modelo 1 (LSTM).**

	precision	recall	f1-score	support
eye opened	0.904	0.924	0.914	1635
eye closed	0.905	0.881	0.893	1344
<b>avg/total</b>	<b>0.904</b>	<b>0.904</b>	<b>0.904</b>	<b>2979</b>

	0	1
0	1184	160
1	125	1510

**Figura 6. Matriz de confusão do modelo 1 (LSTM).**

dades e com *dropout* de 50%. O modelo avaliado obteve acurácia de 93.03%  $\pm$  0.75% (10-fold). As demais métricas são apresentadas na Tabela 2.

**Tabela 2. Resumo da precisão, recall, F1-score para cada classe do modelo 2 (BLSTM).**

	precision	recall	f1-score	support
eye opened	0.940	0.877	0.907	1635
eye closed	0.862	0.932	0.895	1344
<b>avg/total</b>	<b>0.905</b>	<b>0.902</b>	<b>0.902</b>	<b>2979</b>

	0	1
0	1252	92
1	201	1434

**Figura 7. Matriz de confusão do modelo 2 (BLSTM).**

Baseado nos resultados de acurácia (isoladamente) obtidos, pode-se afirmar que os modelos são equivalentes. Considerando as demais métricas e analisando as matrizes de confusão (forma intuitiva de saber como o classificador está se comportando) observa-se que o modelo 1 obteve menos falsos positivos.

O estado da arte para classificação deste conjunto de dados obteve resultados superiores ao obtido ao custo de maior esforço computacional e conseqüentemente maior complexidade [Rösler and Suendermann 2013, Narejo et al. 2016]. Tanto modelos como o *Multilayer Perceptron* (MLP) [Sabancı and Koklu 2015], que obteve seu melhor resultado na configuração 14-7-2, quanto o algoritmo *k-Nearest Neighbors* (kNN) [Sabancı and Koklu 2015], que teve seu melhor resultado quando  $k = 3$ , apresentaram resultados inferiores a 90%. Já os classificadores KStar [Rösler and Suendermann 2013] e KStar+RRF [Hamilton et al. 2015] obtêm resultados superiores a 90% a um custo computacional mais elevado, como aponta [Rösler and Suendermann 2013]. Também foram pesquisados classificadores *Neuro-Fuzzy* [Kim et al. 2016], que apresentaram resultados similares aos de [Rösler and Suendermann 2013, Hamilton et al. 2015]. Por fim, o modelo proposto por [Narejo et al. 2016], utilizando *Stacked AutoEncoders* (SAE), apresenta o melhor desempenho. Ambos os modelos propostos (LSTM e BLSTM) possuem arquitetura simplificada e visam atender minimamente ao compromisso de melhor desempenho e melhor classificação possível.

Além das técnicas utilizadas na literatura para alcançar tais resultados, outros tra-



**Tabela 3. Comparação com os modelos da literatura.**

Modelo	Sucesso na classificação (%)
MLP [Sabancı and Koklu 2015]	56.4
kNN [Sabancı and Koklu 2015]	84.1
LSTM	93.3
BLSTM	93.9
KStar [Rösler and Suendermann 2013]	97.3
KStar+RRF [Hamilton et al. 2015]	96.0
Neuro-Fuzzy [Kim et al. 2016]	97.4
SAE [Narejo et al. 2016]	98.9

balhos relacionados (com diferentes conjuntos de dados) também se destacam, como os que utilizam redes híbridas CNN/LSTM para absorver as características espacial, espectral e temporal do EEG [Bashivan et al. 2015].

## 5. Considerações Finais

Neste artigo, uma abordagem de classificação de séries temporais baseada em redes neurais recorrentes é proposta para a identificação do estado ocular do paciente utilizando seu EEG. A abordagem utilizada focou na utilização de modelos baseados em redes LSTM e suas variantes para efetuar tal classificação. Os resultados obtidos pelos modelos são próximos ou equivalentes aos apresentados pela literatura.

Os resultados finais indicaram que o modelo é aplicável para a classificação de séries temporais de EEG e que o seu desempenho é razoável se comparado aos modelos mais caros computacionalmente.

Trabalhos futuros podem investigar novas abordagens e experimentar técnicas para melhorar as métricas dos modelos apresentados, tendo em vista o compromisso entre custo computacional e desempenho do modelo.

## Referências

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- Acharya, J. N., Hani, A. J., Cheek, J., Thirumala, P., and Tsuchida, T. N. (2016). American clinical neurophysiology society guideline 2: guidelines for standard electrode position nomenclature. *The Neurodiagnostic Journal*, 56(4):245–252.
- Bashivan, P., Rish, I., Yeasin, M., and Codella, N. (2015). Learning representations from eeg with deep recurrent-convolutional neural networks. *arXiv preprint arXiv:1511.06448*.

- Cahn, B. R. and Polich, J. (2006). Meditation states and traits: Eeg, erp, and neuroimaging studies. *Psychological bulletin*, 132(2):180.
- Chollet, F. et al. (2015). Keras. <https://keras.io>.
- Cohen, M. X. (2014). *Analyzing neural time series data: theory and practice*. MIT Press.
- Fonseca, J. P. C. (2016). Fpga implementation of a lstm neural network.
- Graves, A. (2012). *Supervised Sequence Labelling with Recurrent Neural Networks*. Studies in Computational Intelligence 385. Springer-Verlag Berlin Heidelberg, 1 edition.
- Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2017). Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*.
- Hamilton, C. R., Shahryari, S., and Rasheed, K. M. (2015). Eye state prediction from eeg data using boosted rotational forests. In *Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on*, pages 429–432. IEEE.
- Kalchbrenner, N., Danihelka, I., and Graves, A. (2015). Grid long short-term memory. *arXiv preprint arXiv:1507.01526*.
- Kim, Y., Lee, C., and Lim, C. (2016). Computing intelligence approach for an eye state classification with eeg signal in bci. In *Software Engineering and Information Technology: Proceedings of the 2015 International Conference on Software Engineering and Information Technology (SEIT2015)*, pages 265–270. World Scientific.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.
- Malmivuo, J. and Plonsey, R. (1995). *Bioelectromagnetism: principles and applications of bioelectric and biomagnetic fields*. Oxford University Press, USA.
- Narejo, S., Pasero, E., and Kulsoom, F. (2016). Eeg based eye state classification using deep belief network and stacked autoencoder. *International Journal of Electrical and Computer Engineering (IJECE)*, 6(6):3131–3141.
- Northrop, R. B. (2012). *Analysis and Application of Analog Electronic Circuits to Biomedical Instrumentation, Second Edition*. Biomedical engineering series (Boca Raton Fla.). CRC Press, 2nd ed edition.
- Rösler, O. and Suendermann, D. (2013). A first step towards eye state prediction using eeg. *Proc. of the AIHLS*.
- Sabancı, K. and Koklu, M. (2015). The classification of eye state by using knn and mlp classification models according to the eeg signals. *International Journal of Intelligent Systems and Applications in Engineering*, 3(4):127–130.
- Wang, T., Guan, S.-U., Man, K. L., and Ting, T. (2014). Eeg eye state identification using incremental attribute learning with time-series classification. *Mathematical Problems in Engineering*, 2014.
- Zaremba, W., Sutskever, I., and Vinyals, O. (2014). Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.