



## **ANÁLISE COMPARATIVA DE REDES NEURAIS ARTIFICIAIS E ARIMA PARA PREVISÃO DA GERAÇÃO EÓLICA NO ESTADO DE CEARÁ**

**José Péricles Freire<sup>1</sup>, Lihki Rubio<sup>2</sup>, Carlos E. Velasquez<sup>1</sup>**

<sup>1</sup> Departamento de Engenharia Nuclear – Escola de Engenharia Universidade Federal de Minas Gerais (UFMG) Av. Antônio Carlos, 6627 campus Pampulha 31.270-901, Belo Horizonte, MG, Brasil

<sup>2</sup> Departamento de Matemática y Estadística, Universidad del Norte de Barranquilla (UN) Km 5 via, Puerto Colombia, Atlántico, Belo Horizonte, Colombia  
carlosvelcab@nuclear.ufmg.br

**Keywords:** Energia eólica, Redes neurais, Previsão, ARIMA

### **ABSTRACT**

A energia eólica é uma das tecnologias de baixo carbono para geração de energia elétrica. No Brasil existe um grande potencial desta fonte nas regiões nordeste e sul do país. Além disso, diversos têm verificado a complementaridade da geração de energia elétrica da energia eólica à geração hidráulica, isto é que em regime de seca há maior quantidade de ventos e vice-versa. Em 2022 a energia eólica ocupa o segundo lugar como fonte mais usada na matriz elétrica Brasileira com 11.8% equivalente a 25.6 GW, sendo predominantemente hidráulica com 61.9%. Segundo o Plano Nacional de Energia estima-se que em 2050, o Brasil tenha uma capacidade instalada de energia eólica de 110 GW, aumentando assim a capacidade de geração eólica no país. Devido à natureza intermitente da energia eólica, surgem novos desafios para a integração dessa nova fonte na rede elétrica nacional. Para administrar isso de maneira eficiente, a capacidade de prever e analisar tendências é indispensável para saber com exatidão a relação de energia que poderia ser produzida, permitindo assim uma maior precisão na operação do Sistema Interligado Nacional (SIN). Isto poderia levar a realizar escolhas mais precisas em relação ao fornecimento de energia e à administração dos reservatórios. Dentre os principais parâmetros que influenciam um parque eólico temos a direção e a velocidade do vento. Este trabalho propõe estudar diferentes modelos para prever a geração de energia elétrica no estado do Ceará. A partir dos parâmetros mencionados utilizaremos três modelos para realizar previsões no contexto brasileiro, sendo estes o ARIMA, uma rede neural convolucional (CNN) e uma rede neural Long Short-Term Memory (LSTM). A análise comparou os resultados de cada um destes três modelos com os dados reais de velocidade do estado do Ceará para o ano de 2022. Os resultados encontrados mostram um melhor desempenho da rede neural LSTM, possuindo um melhor resultado nas métricas utilizadas para a análise.

### **1. INTRODUÇÃO**

No cenário atual, onde países como China, Índia, Brasil, EUA e países da União Europeia buscam atingir emissões líquidas zero até 2050 [1], a energia eólica vem se apresentando como uma das grandes apostas para a transição energética global focada na redução de dependência de fontes de energia fósseis.

Apesar de ser uma fonte de energia renovável com baixo carbono, a energia eólica enfrenta desafios significativos, especialmente em relação à sua integração nas redes elétricas devido a sua natureza intermitente. Tal característica pode causar congestionamentos nas linhas de transmissão e exige uma coordenação eficiente entre diversos stakeholders [2]. Para lidar com essas dificuldades, são necessárias soluções avançadas, como técnicas de previsão que permitam manter a eficiência e a confiabilidade do sistema.

Integrar uma alta proporção de energia renovável, como ocorre no Brasil, que já em 2022 ocupava a 6ª posição de países com maior capacidade instalada de eólicas [3], apresenta desafios únicos devido



à variabilidade e intermitência dessas fontes. Nesse contexto, as Redes Neurais Artificiais (ANNs) se apresentam como ferramentas fundamentais para a melhoria da previsibilidade de energia eólica, uma vez que oferecem capacidades superiores de modelagem de relações complexas e não lineares quando comparados aos métodos convencionais.

Este trabalho tem como objetivo comparar o desempenho de diferentes métodos preditivos baseados em redes neurais artificiais, focando em sua aplicação na energia eólica para o estado da Ceará, com um método de ajuste já amplamente utilizado para séries temporais, o Modelo Autorregressivo Integrado de Médias Móveis (ARIMA). No estudo implementamos dois tipos distintos de modelos de redes neurais: Long Short-Term Memory (LSTM) [4] e Rede Neural Convolutiva (CNN) [5]. As CNNs podem identificar padrões espaciais dentro dos dados, sendo particularmente úteis para a interpretação de entradas meteorológicas, enquanto as LSTMs lidam com os aspectos temporais dos dados de energia eólica. Ao treinar esses modelos com dados meteorológicos e de vento específicos do Ceará, uma região com grande potencial de energia eólica, este trabalho busca avaliar a melhor abordagem de previsão para o estado do Ceará, melhorando a precisão e a confiabilidade das previsões de energia eólica e contribuindo para a otimização da utilização de energia renovável na região.

## 2. METODOLOGIA

Na realização deste trabalho, utilizamos os dados climáticos provenientes do Instituto Nacional de Meteorologia (BNIM) [6] para o estado do Ceará, abrangendo o período de 2011 a 2021. Foi realizada uma análise dos dados providos para elucidar a dinâmica temporal desses conjuntos. Após esta análise, o modelo ARIMA e os modelos de redes neurais artificiais, abrangendo as arquiteturas CNN e LSTM, foram formulados e treinados utilizando os dados completos da série temporal. O objetivo era aproveitar toda a informação climática disponível para a direção do vento e previsão de velocidade. Após a fase de treinamento, foram geradas previsões para o ano subsequente após o período de treinamento de dados, e esses resultados foram comparados com dados reais de 2022 usando as métricas de avaliação MAPE, MSE, RMSE e  $R^2$ .

### 2.1. DADOS CLIMÁTICOS

Os dados meteorológicos utilizados neste estudo foram fornecidos pelo (BNIM). Os dados de treinamento foram coletados entre os anos de 2011 e 2021, com intervalo horário de dados históricos. Dentre todos os dados fornecidos, os parâmetros utilizados para análise exploratória e construção do modelo incluíram: direção horária do vento, velocidade horária do vento, velocidade máxima do vento na hora anterior, umidade relativa horária, umidade relativa máxima na hora anterior, temperatura horária, temperatura mínima na hora anterior, pressão atmosférica máxima na hora anterior, pressão atmosférica horária e pressão atmosférica mínima na hora anterior. Ao analisarmos a correlação entre as variáveis vimos que a direção do vento possuía uma relação baixa com as outras variáveis utilizadas, para aumentar o suporte dessa variável introduzimos outros dois parâmetros, a projeção vertical da direção do vento e a projeção horizontal da direção do vento. Com exceção da direção do vento, que veio na unidade de grau e foi transformada em radianos, nenhuma variável utilizada para o modelo necessitou de transformações uma vez que chegaram como valores numéricos. Para alimentar o treinamento dos modelos de redes neurais CNN e LSTM foi construída uma matriz que relacionava uma linha dos dados de teste às 23 linhas anteriores resultando em uma matriz de *input* de dimensões (96310, 24, 11).



Conforme mostrado nas Fig 1(a) e Fig 1(b), a velocidade e direção do vento apresentam comportamentos muito diferentes. Para entender melhor o funcionamento destas séries temporais realizamos o teste Dickey-Fuller que confirmou as suposições de estacionariedade de ambas as séries, fornecendo um valor  $p$  que rejeita a hipótese nula de raiz unitária na série temporal, indicando assim estacionariedade. Além deste, testes de hipóteses como Jarque-Bera, teste de assimetria e teste de kurtosis foram aplicados para confirmar a não normalidade na série temporal de velocidade e direção do vento. Esses testes contrastam a hipótese nula de que as séries são normalmente distribuídas, hipótese que é rejeitada com base no valor de  $p$  de cada teste, considerando um nível de significância de 0,05.

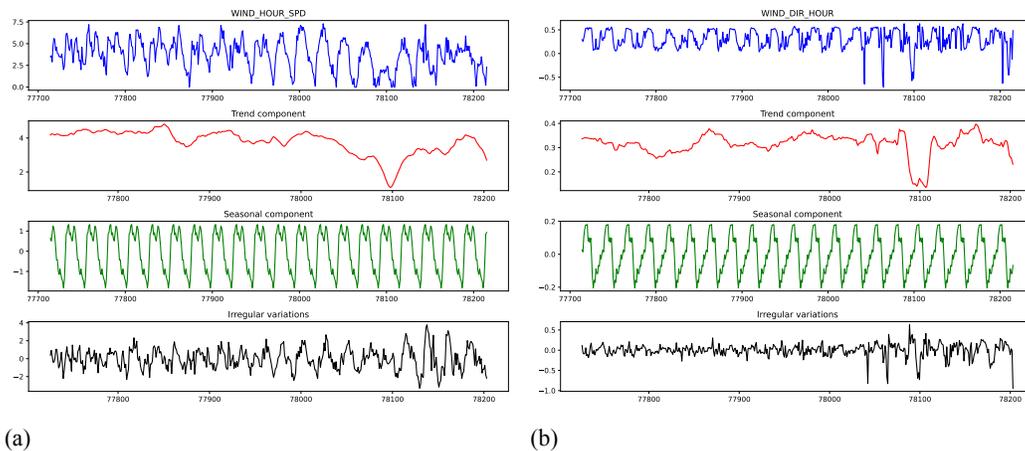


Figura 1. As figuras (a) e (b) apresentam a decomposição aditiva de séries temporais ligadas à velocidade e direção do vento, respectivamente. Esses números mostram componentes de tendência, sazonais e irregulares, com foco nas 500 observações horárias mais recentes.

## 2.2. LONG SHORT TERM MEMORY

LSTM é um tipo de rede neural recorrente (RNN) que permite à rede manter dependências de dados de longo prazo em vários intervalos de tempo. Consiste em uma cadeia de módulos de rede neural repetidos, cada um contendo três portas de controle: a porta de esquecimento, a porta de entrada e a porta de saída. Cada porta compreende uma camada de rede neural sigmóide e uma operação de multiplicação pontual. As camadas sigmóides produzem números entre 0 e 1, indicando a proporção de informações de entrada que devem ser transmitidas. Para dados de série temporal, o LSTM processa uma sequência de vetores de entrada  $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_j, \dots\}$ , onde  $\mathbf{x}_t \in \mathbb{R}^m$  representa um vetor  $m$ -dimensional de leituras para variáveis  $m$  em tempo  $t$ . Várias dessas séries temporais podem ser geradas abrindo uma janela para uma série temporal maior. Embora o LSTM possa lidar com qualquer dado de série temporal, seu desempenho pode variar dependendo da entrada.

Com as novas informações  $\mathbf{x}_t$  no estado  $t$ , o módulo LSTM opera da seguinte forma: Primeiro, ele determina quais informações antigas serão descartadas, gerando um número dentro do intervalo  $[0,1]$ , denotado como



$$(2.1) \quad f_t = \sigma_1(W_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_f),$$

onde  $\mathbf{h}_t$  é a saída no estado  $t - 1$  e  $W_f, b_f$  são as matrizes de peso e viés da porta de esquecimento. Então,  $\mathbf{x}_t$  é processado antes de ser armazenado no estado da célula. O valor  $i_t$  é determinado na porta de entrada, junto com um vetor de valores candidatos  $\tilde{C}_t$  gerado por uma camada tanh, para atualizar o novo estado da célula  $C_t$

$$(2.2) \quad i_t = \sigma_2(W_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_i)$$

$$(2.3) \quad \tilde{C}_t = \tanh(W_c[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_c)$$

$$(2.4) \quad C_t = f_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t,$$

aqui  $(i, i), (b, c)$  são as matrizes de peso e vieses da porta de entrada e do estado da célula de memória, respectivamente. Finalmente, a porta de saída, que é definida por

$$(2.5) \quad o_t = \sigma_3(W_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + b_o)$$

$$(2.6) \quad h_t = o_t \otimes \tanh(C_t),$$

onde  $W_o$  e  $b_o$  são a matriz de peso e polarização da porta de saída, determina uma parte do estado da célula que é gerada. O estado da célula viaja verticalmente por toda a cadeia, preservando internamente as informações sequenciais e permitindo que o LSTM retenha o conhecimento das etapas de tempo subsequentes.[7]–[11].

### 2.3. Modelo Autorregressivo Integrado de Médias Móveis

O modelo Autorregressivo Integrado de Médias Móveis (ARIMA) usa os períodos defasados como variáveis independentes para construir uma equação de regressão. Sua expressão geral é:

$$(2.7) \quad \left(1 - \sum_{i=1}^p \phi_i L^i\right) (1 - L)^d X_t = \left(1 + \sum_{i=1}^q \theta_i L^i\right) \varepsilon_t$$

Em que  $L$  é o operador de defasagem,  $d \in \mathbb{Z}, d > 0$ ,  $p$  e  $q$  representam, respectivamente, o período do modelo autorregressivo e o número de termos do modelo de médias móveis, e  $d$  representa a ordem da diferenciação.

### 2.4. REDE NEURAL CONVOLUCIONAL

As redes neurais convolucionais (CNNs) processam dados de imagem com eficiência usando uma estrutura de feed-forward profunda que inclui blocos de convolução. As CNNs podem extrair automaticamente recursos de dados brutos, tornando-os aplicáveis à previsão de séries temporais, tratando sequências como imagens unidimensionais. Eles oferecem as vantagens dos perceptrons multicamadas, como suporte a entradas e saídas multivariadas e aprendizagem de relações complexas, sem exigir aprendizagem direta de observações defasadas, mas em vez disso concentrando-se em representações relevantes de grandes sequências de entrada [12]–[15].

Uma CNN 1D típica consiste em três camadas principais: camadas convolucionais 1D, camadas de pooling e camadas totalmente conectadas. Além disso, dois parâmetros cruciais são a camada de abandono e a função de ativação.



Uma camada convolucional unidimensional é usada para identificar características em um vetor. O vetor de entrada  $x[p]$ ,  $i = 0, 1, \dots, N - 1$ , é alimentado na camada inicial da CNN. Esta camada emprega vários parâmetros:

- **Filters e Kernels:** Eles produzem mapas de recursos aplicando convoluções aos dados de entrada. A eficácia da detecção de recursos depende do número e tamanho dos kernels. Denotando o kernel de convolução de tamanho  $s$  como  $\kappa[p]$ , a saída de convolução resultante  $\mathcal{C}[p]$  pode ser calculada como

$$(2.8) \quad \mathcal{C}[p] = x[p] \otimes \kappa[p] = \sum_{j=0}^{s-1} \kappa[j]x[p - j]$$

onde  $\otimes$  representa a operação de convolução e  $p = 0, 1, 2, \dots, N - 1$ . Geralmente, os recursos convolvidos na saída da  $l$ ésima camada podem ser expressos como  $\mathcal{C}_k^l = \sigma(b_k^l + \sum_m \mathcal{C}_m^{l-1} \kappa_{km}^l)$ .  $\mathcal{C}_k^l$  representa o  $k$ ésimo recurso na  $l$ ésima camada,  $\mathcal{C}_m^{l-1}$  denota o  $m$ ésimo recurso na  $(l - 1)$ ésima camada,  $\kappa_{km}^l$  é o kernel que conecta o  $k$ ésimo recurso ao  $m$ ésimo recurso,  $b_m^l$  é o viés para esses recursos, e  $\sigma$  é a função de ativação [16].

- **Função de ativação:** um parâmetro crucial no modelo CNN, a função de ativação ajuda a aprender e aproximar relacionamentos contínuos e complexos entre variáveis de rede. Funções de ativação comuns incluem RELU, softmax e sigmoid.
- **Stride:** O valor stride determina o movimento do kernel através dos dados de entrada. O passo mais típico é 1, indicando que o kernel muda uma coluna de dados de entrada a cada etapa.
- **Pooling Layer:** Normalmente posicionada após uma camada convolucional, a camada de pooling visa reduzir o tamanho do mapa de características, diminuindo assim os custos computacionais. Existem várias operações de pooling, como pool máximo, pool médio e pool de soma. Este trabalho utiliza pooling máximo 1D, que envolve deslizar um tamanho de pool especificado e passar sobre a entrada e selecionar o valor máximo de cada região,  $\mathcal{C}_h^l = \max_{e \in \text{mathcal{P}}_h} \mathcal{C}_e^{l-1}$  onde  $\mathcal{P}_h$  representa a região de pooling com índice  $h$ .
- **Camada nivelada e camada eliminada:** A camada nivelada converte os dados de entrada em um vetor unidimensional para a camada totalmente conectada (densa). Para evitar overfitting, que pode ocorrer quando todos os recursos se conectam à camada de nivelamento, uma camada de eliminação é adicionada. Essa camada elimina aleatoriamente alguns neurônios durante o treinamento, reduzindo assim o tamanho e a complexidade do modelo.
- **Camada densa totalmente conectada:** A saída achatada é alimentada na camada densa totalmente conectada, que gera a saída final. Uma função de ativação, como a função RELU descrita anteriormente, é um de seus parâmetros principais.

## 2.5. MÉTRICAS DE AVALIAÇÃO DE DESEMPENHO

Neste estudo, as métricas amplamente reconhecidas MAPE, MSE, RMSE e  $R^2$  foram utilizadas para avaliar o desempenho preditivo dos modelos no conjunto de validação durante a hiperparametrização e mostrar resultados de precisão no conjunto de teste.



$$(2.9) \quad \text{MSE} = \frac{\sum_{i=1}^N [y_i - \hat{y}_i]^2}{N}$$

$$(2.10) \quad \text{RMSE} = \sqrt{\frac{\sum_{i=1}^N [y_i - \hat{y}_i]^2}{N}}$$

$$(2.11) \quad \text{MAPE} = \frac{\sum_{i=1}^N |(y_i - \hat{y}_i)/y_i|}{N} \times 100\%$$

$$(2.12) \quad R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

### 3. RESULTADOS

Os resultados obtidos com cada um dos modelos propostos neste trabalho, incluindo CNN, LSTM e ARIMA, serão apresentados abaixo na Fig 2, representando as últimas 150 observações de cada modelo comparadas com os dados reais sendo Fig 2(a) e Fig 2(b) relativos a velocidade e direção do vento respectivamente, e na Tab 1 que traz os valores de MAPE, MSE, RMSE e  $R^2$  consolidados.

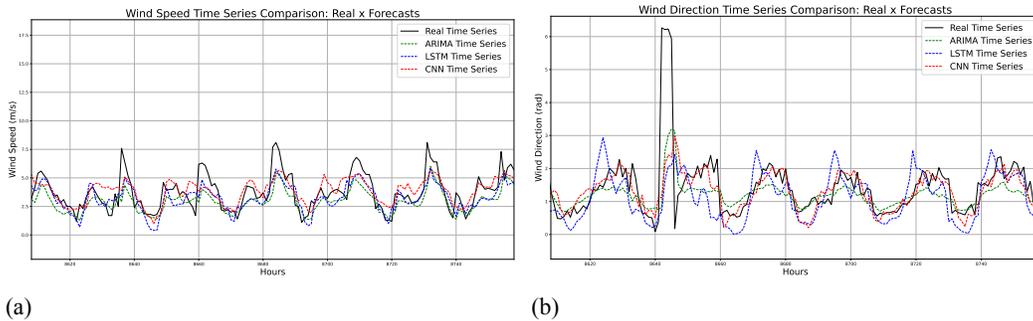


Figura 2. As figuras (a) e (b) representam a previsão da velocidade do vento versus o conjunto de teste e previsão da direção do vento versus o conjunto de teste, respectivamente, para os modelos CNN, LSTM e ARIMA.

De acordo com os dados apresentados na Tab 1, podemos extrair as seguintes informações a respeito das previsões geradas pelos modelos propostos:

- Com relação à velocidade do vento, todos os modelos apresentaram uma capacidade moderada de ajuste, explicando por volta de 60% da variância dos dados de acordo com  $R^2$ . Com base nesse dados, vemos que os modelos com a maior e a menor capacidade de explicar as



		Métricas			
		MAPE	MSE	RMSE	$R^2$
DIR	ARIMA	186.4%	2.57	1.60	0.246
	CNN	236.2%	2.52	1.586	0.262
	LSTM	198.5%	2.50	1.581	0.266
VEL	ARIMA	52.73%	1.096	0.300	0.625
	CNN	84.03%	1.138	0.309	0.611
	LSTM	59.49%	0.889	0.292	0.696

Tabela 1. Tabela de acurácia para previsão de direção e velocidade do vento utilizando modelos CNN, LSTM e ARIMA.

variações dos dados foram respectivamente o LSTM, com  $R^2$  de 0,696, e o CNN, com  $R^2$  de 0,611. Podemos, também ver que, com exceção do MAPE, performado melhor pelo modelo ARIMA com MAPE de 52.73%, o modelo LSTM foi o que possuiu o melhor desempenho nas métricas selecionadas apresentando MSE de 0,889 e RMSE de 0.292 para valores de velocidade que variaram entre 0 m/s e 10 m/s .

- Com relação à direção do vento, todos os modelos apresentaram uma baixa capacidade de ajuste, sendo o melhor modelo capaz de explicar por volta de 26,6% da variância dos dados de acordo com  $R^2$ . Além disso, vemos também um aumento considerável nos valores relativos às métricas de avaliação, sendo o melhor desempenho geral o do modelo LSTM com MSE de 2,50 e RMSE de 1,581, indicam que nenhum dos modelos foi capaz de fazer um bom ajuste. Assim como para a velocidade do vento, o melhor MAPE foi gerado pelo modelo ARIMA com o valor de 186,4%. Essa redução de desempenho pode ser relacionada à incapacidade dos modelos de ajustar a componente de erro da série temporal que, como pode ser visto na Fig 1(b), possui uma magnitude próxima aos valores da série temporal.

#### 4. CONCLUSÃO

Este trabalho examinou o desempenho de diferentes modelos para previsão da velocidade e direção do vento. Os resultados revelam que o modelo LSTM apresentou desempenho superior em ambos os conjuntos de dados, superando os modelos ARIMA e CNN. Em particular, o modelo de rede neural artificial demonstrou melhores resultados na previsão da velocidade do vento, alcançando um erro quadrático médio (MSE) de 88.9% e um coeficiente de determinação  $R^2$  de 0,696, superando os modelos ARIMA ( MSE = 109,6%,  $R^2$  = 0,625) e CNN (MSE = 113,8%,  $R^2$  = 0,611). Além disso, o modelo LSTM produziu resultados promissores na previsão da direção do vento, com um  $R^2$  de 0,266, superando ainda mais os outros modelos. Para melhorar a qualidade dos resultados, a implementação futura de métodos híbridos com redes neurais artificiais, como a Rede Neural Convolutiva Autoencoder LSTM (CNN-ALSTM) e, especialmente, o ARIMA-LSTM, pode ser uma opção viável, combinando a robustez dos modelos estatísticos com a flexibilidade e a capacidade das redes neurais de capturar padrões complexos. Esses resultados poderão ser utilizados para aprimorar a precisão das previsões de energia eólica, contribuindo para uma melhor gestão do Sistema Interligado Nacional, além de otimizar a gestão da geração, transmissão e distribuição de energia elétrica.



## Referências

- [1] International Energy Agency. «Wind Energy». (Accessed on: 8 de setembro de 2024), endereço: <https://www.iea.org/energy-system/renewables/wind>.
- [2] EVWind. «Wind Farms and Grid Integration: Challenges and Solutions». (2023), endereço: <https://www.evwind.es/2023/06/23/wind-farms-and-grid-integration-challenges-and-solutions/92414>.
- [3] Associação Brasileira de Energia Eólica, *Boletim Anual de Geração Eólica 2022*, 2022.
- [4] G. Ponkumar, S. Jayaprakash e K. Kanagarathinam, «Advanced Machine Learning Techniques for Accurate Very-Short-Term Wind Power Forecasting in Wind Energy Systems Using Historical Data Analysis», *Energies*, v. 16, p. 5459, 2023. doi: 10.3390/en16145459. endereço: <https://doi.org/10.3390/en16145459>.
- [5] M. Solas, N. M. Cepeda e J. P. Viegas, «Convolutional Neural Network for Short-term Wind Power Forecasting», em *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, 2019, pp. 1–5.
- [6] Instituto Nacional de Meteorologia. «National Institute of Meteorology». Accessed on: April 16, 2024. (), endereço: <https://portal.inmet.gov.br/dadoshistoricos>.
- [7] A. Sagheer e M. Kotb, «Time series forecasting of petroleum production using deep LSTM recurrent networks», *Neurocomputing*, v. 323, pp. 203–213, 2019.
- [8] F. Shahid, A. Zameer e M. Muneeb, «Predictions for COVID-19 with deep learning models of LSTM, GRU and Bi-LSTM», *Chaos, Solitons & Fractals*, v. 140, p. 110212, 2020.
- [9] Z. Xiang, J. Yan e I. Demir, «A rainfall-runoff model with LSTM-based sequence-to-sequence learning», *Water resources research*, v. 56, n. 1, e2019WR025326, 2020.
- [10] C. Hu, Q. Wu, H. Li, S. Jian, N. Li e Z. Lou, «Deep learning with a long short-term memory networks approach for rainfall-runoff simulation», *Water*, v. 10, n. 11, p. 1543, 2018.
- [11] L. Peng, S. Liu, R. Liu e L. Wang, «Effective long short-term memory with differential evolution algorithm for electricity price prediction», *Energy*, v. 162, pp. 1301–1314, 2018.
- [12] K. Fukushima e S. Miyake, «Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition», *Competition and cooperation in neural nets*, v. 45, pp. 267–285, 1982.
- [13] C. Lang, F. Steinborn, O. Steffens e E. W. Lang, «Applying a 1D-CNN network to electricity load forecasting», em *Theory and Applications of Time Series Analysis: Selected Contributions from ITISE 2019 6*, Springer, 2020, pp. 205–218.
- [14] Y. LeCun, B. Boser, J. Denker et al., «Handwritten digit recognition with a back-propagation network», *Advances in neural information processing systems*, v. 2, 1989.
- [15] D. Hussain, T. Hussain, A. A. Khan, S. A. A. Naqvi e A. Jamil, «A deep learning approach for hydrological time-series prediction: A case study of Gilgit river basin», *Earth Science Informatics*, v. 13, pp. 915–927, 2020.
- [16] M. Saini, U. Satija e M. D. Upadhyay, «Light-weight 1-D convolutional neural network architecture for mental task identification and classification based on single-channel EEG», *arXiv preprint arXiv:2012.06782*, 2020.