

Redes LSTM Otimizadas para Predição de Geração Eólica Diária

Tasso Dênis da Cruz e Silva
Departamento de Engenharia Elétrica
Universidade Federal do Piauí (UFPI)
Teresina, Brasil
tasso.silva@ufpi.edu.br

Pedro Henrique de Souza Portela
Departamento de Engenharia Elétrica
Universidade Federal do Piauí (UFPI)
Teresina, Brasil
pdh290600@ufpi.edu.br

José Maria P. de Menezes Júnior
Departamento de Engenharia Elétrica
Universidade Federal do Piauí (UFPI)
Teresina, Brasil
josemenezesjr@ufpi.edu.br

Resumo—Este trabalho investiga a aplicação de redes neurais artificiais na predição horária da geração de energia em usinas eólicas no estado do Piauí. Utilizou-se um modelo baseado em *Long Short-Term Memory* (LSTM), treinado com dados históricos do Operador Nacional do Sistema Elétrico (ONS), referentes ao período de janeiro a novembro de 2024. O desempenho da LSTM foi comparado com os modelos FTDNN e XGBoost, utilizando métricas de diferentes aspectos do desempenho preditivo. Os modelos foram otimizados com o auxílio de técnicas de AutoML, visando selecionar automaticamente os melhores hiperparâmetros. Os resultados demonstram que a LSTM apresentou desempenho superior na maioria dos critérios avaliados, especialmente na identificação de picos e de oscilações abruptas da série temporal. Esses achados destacam o potencial da LSTM como uma solução robusta e eficiente para predição em ambientes de alta variabilidade, como o setor de geração de energia eólica.

Palavras-chave—Redes neurais; Otimização; Algoritmos Genéticos; predição; Séries Temporais.

I. INTRODUÇÃO

Com o crescimento populacional, Solarin (2020) afirma que a demanda por energia aumenta, enquanto os recursos de combustíveis fósseis, que geram impactos ambientais significativos, tendem a se esgotar ao longo do tempo. Concomitantemente, Qadir et al. (2021) destaca a necessidade urgente de investimentos em fontes de energia limpa, discutindo barreiras e propondo incentivos para acelerar a transição para fontes renováveis, como a energia eólica. Ainda nesse contexto, Orsini (2018) ressalta a importância de analisar e identificar padrões de variabilidade de longo prazo na energia eólica, buscando regiões que proporcionem um fornecimento mais confiável e, assim, contribuam para a segurança da matriz energética brasileira.

O Brasil tem direcionado esforços para fomentar o investimento em energias renováveis, tanto na modalidade *onshore* quanto *offshore* da energia eólica, por meio de iniciativas públicas, como o Programa de Incentivo às Fontes Alternativas de Energia Elétrica (PROINFA), e privadas (Martins et al., 2023). Como resultado, o país manteve, pelo segundo ano consecutivo, a terceira maior capacidade instalada de energia eólica em 2021, ficando atrás apenas da China e dos Estados Unidos (GWEC, 2023).

Dados de 2023 da Associação Brasileira de Energia Eólica (ABEEólica) indicam uma produção nacional de energia eólica

de 26 GW, com predominância na região Nordeste. Em 2021, essa região foi responsável por aproximadamente 90,3% da produção nacional, correspondendo a 70,48 TWh (ABEEólica, 2023). De acordo com Lordêlo and Sant'Anna (2024), investir em sistemas de predição de geração eólica pode trazer benefícios como a redução dos desequilíbrios entre geração e consumo, mitigação de cortes de fornecimento, maior eficiência no planejamento e operação dos sistemas, aumento da segurança das equipes de manutenção e minimização dos impactos de condições meteorológicas extremas.

Diante desse cenário, este estudo analisa a geração média diária de energia eólica no estado do Piauí, com o objetivo de identificar tendências de produção e realizar predições utilizando modelos neurais baseados em memória de curto prazo e redes com múltiplas camadas ocultas. Considerando a posição de destaque do Brasil entre os maiores produtores mundiais de energia eólica, pesquisas voltadas para essa matriz energética têm potencial para gerar contribuições relevantes no avanço da produção e gestão de energia limpa no país.

Nesse contexto, propõe-se a comparação de três abordagens baseadas em aprendizado de máquina: duas redes neurais artificiais, *Long Short-Term Memory* (LSTM) e *Focused Time Delay Neural Network* (FTDNN), e o modelo *eXtreme Gradient Boosting* (XGBoost), que utiliza um conjunto de árvores de decisão. A escolha se justifica pela complementaridade de suas arquiteturas diante das exigências da tarefa de predição em um cenário de alta variabilidade. Essas técnicas foram aplicadas à tarefa de predição da geração diária de energia de usinas eólicas localizadas no estado do Piauí, utilizando dados históricos disponibilizados pelo Operador Nacional do Sistema Elétrico (ONS), referentes ao ano de 2024. O objetivo central é identificar qual abordagem apresenta o menor erro de predição e, portanto, maior precisão na estimativa de valores futuros.

Além da comparação entre arquiteturas de redes neurais, este trabalho também incorpora uma etapa de otimização dos hiperparâmetros dos modelos, atividade alinhada à área de AutoML (*Automated Machine Learning*), que visa automatizar a seleção e ajuste de modelos. A escolha adequada de hiperparâmetros, como número de neurônios, taxa de aprendizado e tamanho da janela temporal, pode impactar significativamente o desempenho de modelos de predição, sobretudo em proble-

mas desafiadores como séries temporais de alta variabilidade, características da geração de energia eólica.

A otimização de hiperparâmetros para predição de séries temporais é uma tarefa notoriamente complexa, uma vez que envolve espaços de busca amplos, alta dependência sequencial dos dados e múltiplos critérios de avaliação. Para lidar com essas dificuldades, este estudo adota duas abordagens distintas: o método de *grid search*, que realiza uma busca exaustiva em um espaço pré-definido de combinações, e a utilização de algoritmos genéticos, técnica inspirada em processos evolucionários naturais, visando explorar de forma mais eficiente regiões promissoras do espaço de soluções.

Assim, a principal contribuição deste trabalho é a avaliação integrada do impacto da escolha da arquitetura da rede neural e da otimização automática de seus hiperparâmetros, aplicadas à predição de geração eólica no Piauí. Esta abordagem oferece uma estratégia mais robusta e adaptável para aplicações reais no setor elétrico, possibilitando predições mais precisas em cenários caracterizados por elevada incerteza e variabilidade.

Este artigo está organizado da seguinte maneira: a próxima seção apresenta as principais características dos modelos empregados; em seguida, detalha-se a metodologia adotada para o processamento e predição dos dados; por fim, são discutidos os resultados e expostas as conclusões do estudo.

II. REVISÃO DE LITERATURA

Nesta seção, são apresentados os principais modelos dinâmicos, arquiteturas neurais e técnicas modernas de aprendizado de máquina utilizados na tarefa de predição de séries temporais, com ênfase principal em aplicações na geração de energia eólica.

A. Rede Neural Dinâmica

O problema de predição não-linear de séries temporais é geralmente colocado na forma de um problema de aproximação de funções. Assim, o interesse em usar a Rede *MultiLayer Perceptron* (MLP) para predição está fundamentado justamente na capacidade de aproximação universal desta arquitetura neural. Além disto, para ter bom desempenho em tarefas de predição de séries temporais, a rede MLP deve também ser capaz de representar a dinâmica temporal (relações de causa-e-efeito) do processo não-linear que gerou a série temporal e que está implicitamente representada nesta.

Uma rede neural capaz de modelar a dinâmica de um processo, a partir de uma série temporal, é chamada de rede neural dinâmica, caso contrário, a rede é dita estática. Uma rede neural pode ser concebida como dinâmica ou ser tornada dinâmica a partir de uma rede estática. As arquiteturas de redes neurais dinâmicas são, em sua grande maioria, extensões da rede MLP, que, por sua vez, é originalmente uma rede estática, ou seja, voltada para o processamento de dados estáticos provenientes de sistemas sem memória.

A rede MLP e suas variantes dinâmicas são tornadas sensíveis à estrutura temporal dos sinais portadores de informações através da inclusão de mecanismos de memória

de curta duração (*short-term memory*, STM). Estes mecanismos são responsáveis por manter a informação temporal disponível por vários instantes de tempo, a fim de que a rede MLP possa manipulá-la e armazená-la adequadamente em seus pesos sinápticos. Uma forma simples de inserir memória de curta duração dá-se através de atrasadores (*time delays*) ou laços de realimentação (*feedback loops*), que podem ser inseridos tanto interna quanto externamente à rede.

A rede neural dinâmica formada pela introdução de atrasadores na entrada de uma rede MLP estática é conhecida pela sigla FTDNN (*Focused Time Delay Neural Network*) (Principe et al., 2000), sendo aqui chamada simplesmente de Rede MLP com Atrasadores na Entrada, mostrada na Fig. 1. Assim, como a rede MLP, que lhe dá origem, a FTDNN é uma rede *feedforward* multicamadas cujos pesos sinápticos e limiares são ajustados de acordo com o algoritmo *backpropagation*.

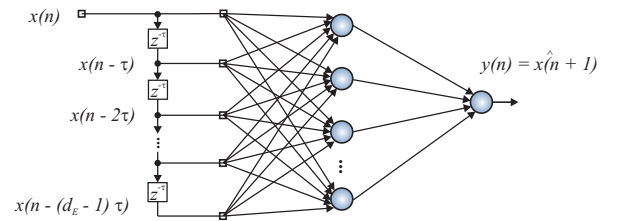


Figura 1: Arquitetura genérica de uma rede FTDNN de uma camada oculta.

B. Modelo Extreme Gradient Boosting

O *Extreme Gradient Boosting* (XGBoost) é um algoritmo baseado em árvores de decisão proposto por Chen and Guestrin (2016), utilizado em tarefas de classificação e regressão. Ele se baseia no princípio do *boosting*, uma técnica de aprendizado que combina sequencialmente modelos fracos, neste caso, árvores de decisão, com o objetivo de melhorar progressivamente o desempenho preditivo do conjunto.

Embora não tenha sido originalmente projetado para dados sequenciais, o XGBoost é frequentemente empregado como base comparativa em tarefas de séries temporais, desde que as dependências temporais sejam explicitamente modeladas por meio de janelas deslizantes ou variáveis defasadas.

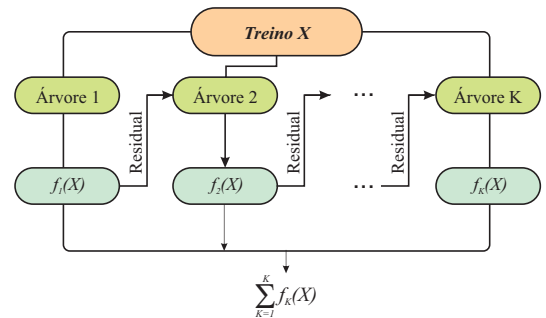


Figura 2: Esquema do treinamento em etapas do XGBoost.

A Fig. 2 representa o funcionamento interno do XGBoost, destacando seu processo de treinamento sequencial por meio

da construção de múltiplas árvores de decisão. Inicialmente, o conjunto de dados $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ é utilizado para treinar a primeira árvore f_1 , que gera uma predição $\hat{y}_1$. Na sequência, a segunda árvore f_2 é ajustada para minimizar os resíduos da predição anterior, ou seja, os erros entre y_i e $\hat{y}_1$. Esse processo continua até a árvore f_K , gerando a predição final $\hat{y}_i = \sum_{k=1}^K f_k(x_i)$.

Cada função f_k representa uma árvore de decisão que contribui com um valor de saída a ser somado ao modelo final. O treinamento é guiado pelo gradiente da função de perda, e a regularização é aplicada sobre os pesos dos nós-folha (w_j) para evitar sobreajuste. A figura também destaca a distinção entre os nós internos (de decisão) e os nós-folha (de saída), além da composição aditiva da predição.

Segundo Natekin and Knoll (2013), o XGBoost destaca-se pela capacidade de lidar com dados ausentes, execução paralela eficiente e suporte a diversas funções de perda. Sua flexibilidade na calibração de hiperparâmetros, como profundidade das árvores, número de estimadores e taxa de aprendizado, permite ajustes interessantes ao problema analisado.

C. Rede Neural Long Short-Term Memory

As redes *Long Short-Term Memory* (LSTM) constituem uma variação das redes neurais recorrentes (RNN), inicialmente propostas por Hochreiter and Schmidhuber (1997). Essa arquitetura foi desenvolvida para lidar com sequências temporais de forma eficiente, superando limitações de RNNs tradicionais, como o desaparecimento do gradiente em longas dependências temporais.

A LSTM processa uma entrada da série temporal por vez, mantendo duas estruturas principais: o estado interno da célula e o vetor oculto (estado de saída). Ambos são inicialmente zerados e atualizados a cada novo instante da sequência. Por meio de mecanismos de controle internos, chamados *gates*, a rede decide quais informações devem ser mantidas, descartadas ou repassadas. Esses portões regulam a entrada de novas informações, o esquecimento de conteúdos passados e a saída para o próximo passo da sequência.

As LSTMs são especialmente indicadas para tarefas de predição e modelagem de séries temporais, graças à sua capacidade de capturar dependências de longo prazo (Van Houdt et al., 2020; Staudemeyer and Morris, 2019). Essa habilidade é viabilizada por sua arquitetura especializada, na qual cada célula possui três portas principais: a porta de entrada, que regula a atualização do estado interno; a porta de esquecimento, que decide o que deve ser descartado; e a porta de saída, que define o que será transmitido à próxima etapa da rede. A Fig. 3 ilustra essa estrutura.

Matematicamente, conforme descrito por Wang et al. (2019), o funcionamento de uma célula LSTM é definido pelas Eq. (1) a (5). Nelas, x_t é a entrada no tempo t , h_{t-1} representa o estado oculto anterior, e c_t , o estado interno. As portas i_t , f_t e o_t correspondem, respectivamente, às funções de entrada, esquecimento e saída, responsáveis por controlar o fluxo de informação.

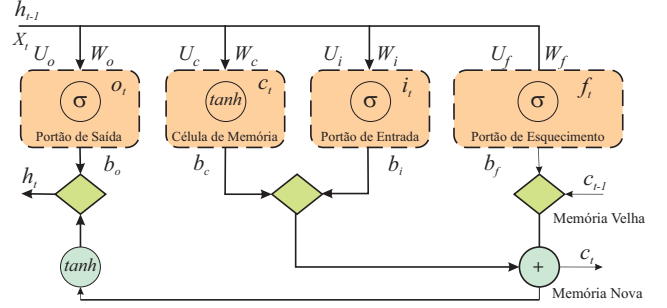


Figura 3: Esquema de uma célula LSTM.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i), \quad (1)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f), \quad (2)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad (3)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(W_c[h_{t-1}, x_t] + b_c), \quad (4)$$

$$h_t = o_t \cdot \tanh(c_t). \quad (5)$$

Essas expressões definem, a cada passo temporal, como a célula decide o que armazenar, esquecer ou repassar. A função sigmoide σ atua como filtro de controle, atribuindo pesos entre 0 e 1 para cada porta.

Redes dinâmicas convencionais, sem a arquitetura especializada da LSTM, apresentam limitações em capturar dependências temporais longas, devido à atenuação progressiva dos sinais ao longo do tempo, fenômeno conhecido como *vanishing gradient* (Noh, 2021; Haykin, 2001). A arquitetura LSTM tende a superar essa limitação ao preservar informações relevantes no estado interno por períodos mais longos, tornando-se especialmente adequada para modelagem de séries temporais com padrões complexos e dependência de longo prazo.

D. Otimização de Hiperparâmetros

A escolha adequada dos hiperparâmetros é uma etapa crítica no desenvolvimento de modelos de aprendizado de máquina, pois influencia diretamente o desempenho preditivo. Métodos tradicionais de otimização, como a busca em grade (*Grid Search*) e a busca aleatória (*Random Search*), embora amplamente utilizados, tornam-se ineficientes em espaços de busca extensos ou quando o custo de avaliação de cada configuração é elevado.

Nos últimos anos, abordagens mais avançadas têm sido exploradas no contexto do AutoML (Aprendizado de Máquina Automatizado), que visa automatizar processos como seleção de modelos e ajuste de hiperparâmetros. O AutoML utiliza estratégias mais eficientes, como otimização bayesiana, algoritmos evolutivos e pesquisa em árvores de Parzen, permitindo uma exploração mais eficaz de grandes espaços de configuração (Hutter et al., 2019; Feurer et al., 2015). Isso torna o processo mais escalável e acessível, inclusive

para usuários com pouca familiaridade com aprendizado de máquina.

Neste estudo, especificamente, dentre as estratégias mencionadas, foi utilizado o Optuna, uma biblioteca de código aberto em Python para otimização de hiperparâmetros, utilizando o Estimador de Parzen Estruturado em Árvore (TPE) para explorar espaços de hiperparâmetros de forma eficiente (Akiba et al., 2019). Por meio de estudos e ensaios, ele avalia configurações com base em uma função objetivo, ajustando a busca dinamicamente com modelagem bayesiana e poda de ensaios pouco promissores. Integra-se com frameworks como TensorFlow, PyTorch e XGBoost, sendo aplicado em previsão de doenças cardiovasculares (Ali, 2024), energia eólica (Jafari, 2024) e mapeamento de mangues (Chen, 2024). Oferece visualizações para análise de hiperparâmetros, como em estudos de XCO₂ (Li, 2024), e suporta otimização distribuída, otimizando modelos em diversos domínios.

Os modelos analisados neste estudo (XGBoost, FTDNN e LSTM) apresentam diferentes estratégias para lidar com séries temporais. O XGBoost, baseado em árvores de decisão e *boosting*, destaca-se pela robustez e eficiência, sendo ajustado por hiperparâmetros como profundidade das árvores, número de estimadores e taxa de aprendizado. A FTDNN, com sua arquitetura *feedforward* e janelas temporais, é uma opção mais simples e estável para padrões temporais bem definidos, com ajustes em número de camadas, neurônios e funções de ativação. Já a LSTM, especializada em capturar dependências de longo prazo, utiliza mecanismos de memória e portas internas, e sua performance depende do ajuste de hiperparâmetros como tamanho da janela, unidades ocultas, taxa de *dropout* e número de épocas.

III. METODOLOGIA

Esta seção descreve os procedimentos adotados para construção, treinamento e avaliação dos modelos neurais aplicados à predição da geração diária de energia eólica.

As implementações foram realizadas em linguagem Python, utilizando a biblioteca Keras com backend TensorFlow. Os modelos foram desenvolvidos por meio da API *Sequential*, com uso de camadas *Dense* para redes MLP e camadas *LSTM* para redes recorrentes. O treinamento foi conduzido com séries temporais históricas de geração elétrica, fornecidas pelo Operador Nacional do Sistema Elétrico (ONS), conforme detalhado nas subseções seguintes.

A. Pré-processamento dos Dados

As observações de geração de energia eólica foram obtidas do portal de dados abertos do Operador Nacional do Sistema Elétrico (ONS), totalizando 111.745 amostras relativas ao período de janeiro a dezembro de 2024.

A Tab. I apresenta os valores médios mensais, máximos e o desvio padrão da geração observada. Nota-se um aumento significativo da média de geração a partir de maio, com pico entre os meses de agosto e setembro. Esse padrão sazonal está relacionado ao fenômeno regional conhecido como “safra dos ventos”, caracterizado por ventos mais intensos e constantes

no segundo semestre do ano, especialmente no Nordeste brasileiro.

Tabela I: Valores médios mensais de geração eólica no Piauí.

Mês	Geração Média (MWmed)	Geração Máxima	Desvio Padrão
Jan	51,35	644,43	72,24
Fev	108,23	675,06	128,83
Mar	73,89	661,88	107,54
Abr	111,80	661,43	121,13
Mai	177,10	662,16	146,33
Jun	196,19	685,66	145,72
Jul	212,92	730,02	153,78
Ago	208,06	722,02	155,55
Set	200,70	928,70	161,83
Out	181,87	927,26	193,74
Nov	163,75	921,11	182,22
Dez	135,46	919,38	150,23

Em relação aos valores ausentes, foi adotada uma estratégia de imputação baseada na média dos valores vizinhos mais próximos. Esse procedimento foi necessário devido à presença de lacunas nos dados históricos, um dos principais desafios enfrentados na análise de séries meteorológicas no Brasil (Curtarelli et al., 2014).

Após o pré-processamento, procedeu-se à análise exploratória do comportamento da série temporal. De acordo com Lustosa et al. (2004), as séries temporais podem ser influenciadas por fatores diversos, como mudanças macroeconômicas, evolução tecnológica, variações naturais e eventos imprevisíveis. Esses elementos se refletem nos componentes estruturais da série, que incluem tendência, sazonalidade e variações irregulares, conforme ilustrado na Fig. 4.

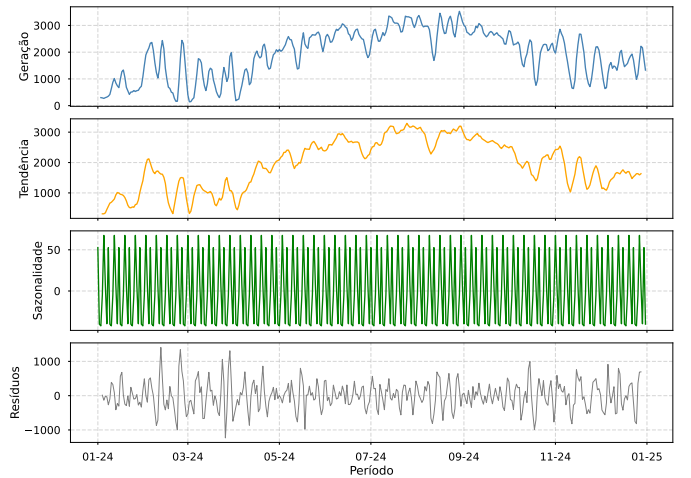


Figura 4: Decomposição da série temporal de geração eólica em tendência, sazonalidade e componente irregular.

A Fig. 4 mostra a decomposição aditiva da série de geração eólica em tendência, sazonalidade e resíduos. Observa-se um crescimento gradual da geração até setembro, seguido de queda. A tendência (segundo painel) confirma esse padrão, enquanto a componente sazonal revela oscilações regulares de alta frequência, associadas a variações diárias na velocidade

dos ventos. O componente residual (quarto painel) exibe ruído com média próxima de zero e leve aumento na dispersão nos extremos da série. De modo geral, a decomposição evidencia uma estrutura com tendência e sazonalidade bem definidas, justificando o uso de modelos como as redes LSTM, que são adequadas para lidar com dependências temporais complexas.

Para avaliar a estacionariedade da série temporal, foi aplicado o Teste de Dickey-Fuller Aumentado (ADF), conforme proposto por Elliott et al. (1992). Esse teste avalia a hipótese nula de que a série possui raiz unitária, ou seja, não é estacionária. A estatística de teste foi de -4,728610, com um valor-p associado de 0,000074. Considerando um nível de significância de 5% (0,05), o valor-p é suficientemente baixo para rejeitar a hipótese nula, indicando que a série é estacionária. Portanto, não foi necessário aplicar diferenciação aos dados antes do ajuste dos modelos preditivos.

B. Aplicação dos Modelos

Nas previsões com um passo à frente (*one-step-ahead*), os modelos preditivos devem estimar o próximo valor de uma série temporal sem realimentá-lo para o vetor de entrada do modelo. Quando o objetivo é uma previsão com horizonte mais longo, conhecida como previsão com múltiplos passos à frente (*multistep-ahead* ou *long-term prediction*), a saída do modelo precisa ser realimentada ao regressor de entrada por um número fixo, porém finito, de passos de tempo (Sorjamaa et al., 2007). A previsão com múltiplos passos à frente é consideravelmente mais complexa do que a previsão de um passo, sendo amplamente reconhecida como uma tarefa na qual os modelos de redes neurais artificiais desempenham um papel relevante, especialmente os modelos recorrentes.

Neste estudo, a série temporal foi dividida de forma sequencial, com treinamento de 01/01/2024 a 30/11/2024, e testes no mês de dezembro de 2024. Após normalização, os dados foram organizados em janelas deslizantes (*sliding window*) para entrada no modelo. O processo de tuning de hiperparâmetros envolveu a variação do tamanho da janela, número de unidades, taxa de dropout, taxa de aprendizado, regularização L2 e tamanho do batch, buscando a minimização do erro absoluto médio (MAE).

O processo foi realizado utilizando previsão recursiva, onde a previsão de cada passo é reinserida na janela de entrada para estimar os valores seguintes, com correções periódicas baseadas nos dados reais a cada intervalo fixo.

Em tarefas de previsão, o uso de janelas deslizantes (*sliding windows*) permite transformar a série temporal em um problema de aprendizado supervisionado. Para cada instante n , constrói-se uma janela de tamanho w (hiperparâmetro), composta pelos w valores anteriores normalizados, que servem como vetor de entrada do modelo. O valor correspondente ao instante n é utilizado como rótulo (*label*) de saída. Esse processo é aplicado de forma contínua aos conjuntos de treinamento e teste, gerando pares entrada-saída compatíveis com a abordagem supervisionada.

A primeira arquitetura analisada neste estudo é a rede LSTM, implementada com a biblioteca *Keras* no ambiente

Google Colaboratory. Após o pré-processamento, os dados históricos foram utilizados para treinar uma rede LSTM multi-camadas construída com a API *Sequential*. A saída h_t da célula LSTM foi propagada sequencialmente como entrada para os passos seguintes da série temporal, mantendo a coerência temporal do modelo.

A configuração do modelo inclui duas camadas ocultas com 32 neurônios cada, ativação *ReLU*, seguidas por camadas de *dropout* com taxa de 0,25 para regularização. A saída é produzida por uma camada densa com 5 neurônios, responsáveis pela previsão dos cinco valores seguintes da série. O treinamento foi conduzido utilizando o otimizador *Adam*, um algoritmo baseado em gradiente estocástico com estimativas adaptativas de momentos de ordem inferior (Kingma, 2014). O modelo foi treinado por até 50 épocas com *early stopping*, utilizando *batch size* determinado empiricamente via otimização bayesiana. As janelas deslizantes de entrada usaram um número de *lags* entre 1 e 24, escolhido automaticamente para capturar dependências temporais de curto e médio prazo.

Para fins comparativos, foi implementada uma rede FTDNN, utilizando a biblioteca *Scikit-learn*. O modelo foi configurado para processar janelas deslizantes de 10 *timesteps*, com normalização dos dados por meio da técnica *Min-Max Scaling*, assegurando uniformidade nas entradas da rede.

A otimização dos hiperparâmetros foi conduzida com o framework *Optuna* (com algoritmos bayesianos e estratégias de pruning), avaliando 10 configurações distintas. As variações incluíram: 1 a 3 camadas ocultas, número de neurônios entre 1 e 127, taxa de aprendizado entre 0,00001 e 0,01, *batch size* de 32 a 128, número de épocas entre 100 e 500, e função de ativação *ReLU*. A função objetivo adotada foi a minimização do erro quadrático médio (MSE) no conjunto de validação. Após o treinamento com os melhores hiperparâmetros, as previsões foram reescaladas.

Além das redes neurais, foi implementado um modelo baseado no algoritmo XGBoost. A seleção dos hiperparâmetros foi realizada por meio do framework *Optuna*, com a execução de 10 experimentos para explorar diferentes combinações de configuração. Os parâmetros otimizados incluíram: $n_estimators$ (entre 50 e 300), max_depth (de 3 a 10) e $learning_rate$ (de 0,01 a 0,3). A função objetivo adotada foi a minimização do erro quadrático médio (MSE) no conjunto de teste, garantindo maior precisão na tarefa de previsão.

Os modelos finais foram configurados com os hiperparâmetros definidos pelos otimizadores, resultando em estruturas ajustadas às particularidades da série de geração eólica. Com os modelos treinados e as previsões reescaladas, procede-se à avaliação de desempenho com base em métricas consagradas, conforme descrito na próxima seção.

C. Critérios de Avaliação

Para avaliar o desempenho dos modelos de previsão da geração eólica, foram empregadas métricas consagradas na literatura de regressão e séries temporais: MAE, MSE, RMSE, MAPE, R^2 , NMSE e U_{Theil} . Sejam $\hat{y}(n)$ os valores preditos,

$y(n)$ os valores reais, $\bar{y}$ a média de $y(n)$, e H é o comprimento da sequência de exemplos preditos. As expressões correspondentes são:

$$MAE = \frac{1}{H} \sum_{n=1}^H |\hat{y}(n) - y(n)|, \quad (6)$$

$$MSE = \frac{1}{H} \sum_{n=1}^H (\hat{y}(n) - y(n))^2, \quad (7)$$

$$RMSE = \sqrt{MSE}, \quad (8)$$

$$MAPE = \frac{1}{H} \sum_{n=1}^H \left| \frac{\hat{y}(n) - y(n)}{y(n)} \right|, \quad (9)$$

$$R^2 = 1 - \frac{\sum_{n=1}^H (\hat{y}(n) - y(n))^2}{\sum_{n=1}^H (\bar{y} - y(n))^2}, \quad (10)$$

$$NMSE = \frac{\sum_{n=1}^H (\hat{y}(n) - y(n))^2}{\sum_{n=1}^H (\bar{y} - y(n))^2}, \quad (11)$$

$$U_{Theil} = \sqrt{\frac{\sum_{n=2}^H (\hat{y}(n) - y(n))^2}{\sum_{n=2}^H (y(n-1) - y(n))^2}}. \quad (12)$$

As métricas adotadas avaliam diferentes aspectos do desempenho preditivo. MAE, MSE e RMSE quantificam o erro absoluto e quadrático, com RMSE penalizando mais os grandes desvios. O MAPE expressa o erro em termos percentuais, facilitando a comparação entre séries. O coeficiente R^2 indica o quanto da variância é explicada pelo modelo. Métricas como NMSE e U_{Theil} oferecem avaliações normalizadas, úteis em comparações relativas. Juntas, essas métricas fornecem uma visão abrangente da qualidade das previsões.

IV. RESULTADOS

Nesta seção são apresentados e discutidos os resultados obtidos na tarefa de previsão da geração de energia eólica com horizonte de um passo à frente, utilizando os modelos LSTM, XGBoost e FTDNN. O objetivo da análise é comparar o desempenho dessas abordagens em relação à capacidade de prever com precisão os valores futuros da série temporal, com base em dados históricos disponibilizados pelo Operador Nacional do Sistema Elétrico (ONS).

Os resultados apresentados foram obtidos com modelos otimizados automaticamente por meio do AutoML, que selecionou os melhores hiperparâmetros de forma eficiente. Esse processo foi especialmente vantajoso em arquiteturas mais complexas, como a LSTM, ao reduzir o tempo de experimentação e evitar ajustes manuais exaustivos. Dessa forma, os modelos aqui avaliados representam as melhores versões encontradas para cada técnica após a etapa de ajuste automático.

Cada modelo foi treinado e testado com os mesmos dados pré-processados, seguindo os critérios estabelecidos na metodologia. As métricas de avaliação foram calculadas para aferir a qualidade das previsões. A Tabela II consolida os

resultados, evidenciando que o modelo LSTM superou as demais abordagens em todos os indicadores, apresentando erros mais baixos e maior capacidade de explicação da variabilidade da série. Esses resultados indicam que a arquitetura LSTM, ao capturar dependências temporais de curto e longo prazo, oferece vantagens significativas na modelagem de séries com oscilações frequentes, como é o caso da geração eólica.

Tabela II: Métricas de desempenho.

Métricas	LSTM	XGBoost	FTDNN
MAE	66,60	165,21	187,70
MSE	8621,05	53591,03	53840,06
RMSE	92,84	231,48	232,03
MAPE	5,07	12,63	12,45
R^2	0,9748	0,8581	0,8239
NMSE	0,0281	0,1418	0,1760
U_{Theil}	0,0289	0,0709	0,0768

A Figura 5 complementa essa análise ao comparar, por meio de um gráfico boxplot, a distribuição dos valores de NMSE entre os modelos, destacando a consistência e o desempenho superior da LSTM. A vantagem da LSTM tende a decorrer de sua estrutura interna, baseada em mecanismos de memória que preservam informações relevantes ao longo da sequência temporal. A cada etapa, a rede mantém um estado interno capaz de considerar parâmetros anteriormente processados, caracterizando a presença de uma memória de longo prazo (Seidel, 2022). Essa capacidade permite à LSTM acompanhar oscilações sazonais e eventos pontuais da série com maior precisão.

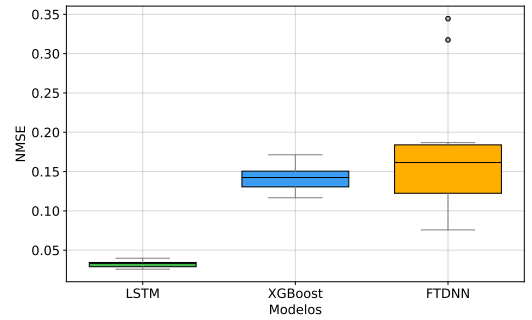


Figura 5: Boxplot comparando os valores de NMSE obtidos pelos modelos LSTM, XGBoost e FTDNN.

Em contraste, a FTDNN mostrou-se menos eficaz para capturar variações bruscas ou prever valores de pico nas oscilações diárias. Sua resposta é limitada a valores próximos da média móvel da série, como ilustrado na Fig. 6. Isso ocorre porque, por não possuir mecanismos explícitos de memória, a FTDNN apresenta dificuldades em modelar dependências temporais mais longas.

A rede XGBoost, embora tenha superado levemente a FTDNN em desempenho, também se mostrou significativamente inferior ao modelo LSTM. Essa limitação é visível na Fig. 7, onde se nota uma defasagem nas previsões em relação

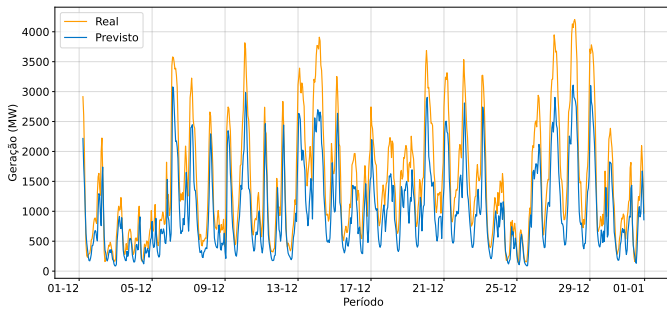


Figura 6: Resultado da predição horária com a rede FTDNN.

aos valores reais, principalmente nos momentos de transição e nos extremos da série.

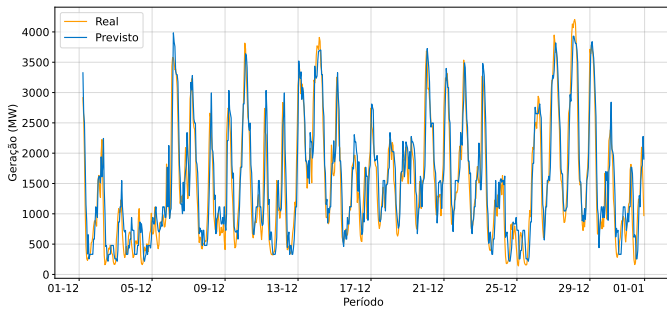


Figura 7: Resultado da predição horária com a XGBoost.

O valor de R^2 (0,9748) alcançado pela LSTM indica que o modelo explica mais de 97% da variabilidade da série, enquanto o MAPE de 5,07% revela uma precisão elevada nas predições. Já os baixos valores de U_{Theil} (inferiores a 0,03) reforçam a qualidade do ajuste em comparação com os modelos alternativos.

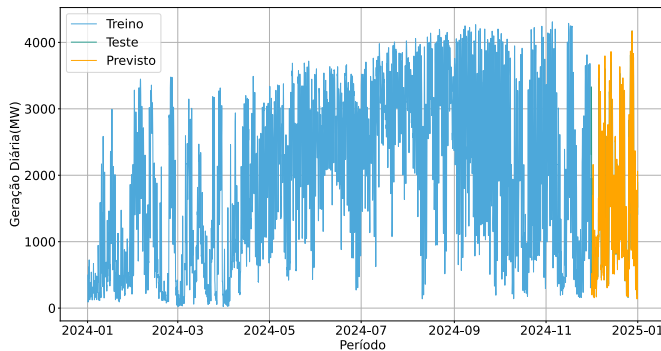


Figura 8: Resultado para toda a série com a rede LSTM.

Para visualização, a Fig.8 mostra que os dados preditos seguiram a densidade de valores dessa série temporal e acompanharam, com uma boa curva, os dados de teste. Além disso, para o horizonte de predição de 31 dias, a rede LSTM obteve

uma boa performance, acompanhando as oscilações dos dados, Fig.9, causados por fatores externos como por exemplo, a redução da intensidade do vento em horários específicos do dia, causando uma redução na geração das usinas.

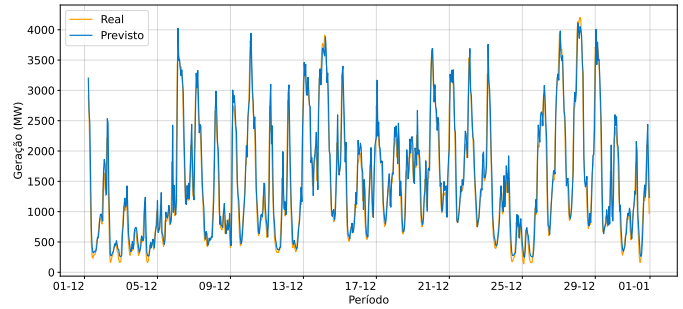


Figura 9: Resultado da predição horária com a rede LSTM.

Embora o LSTM apresente maior complexidade, essa característica permite ao modelo capturar padrões mais sutis e variações abruptas na série. Oscilações diárias, com picos máximos e mínimos, ajudam no aprendizado de dependências temporais, favorecido pela memória interna da LSTM.

O uso do AutoML na etapa de otimização de hiperparâmetros contribuiu significativamente para a melhoria dos resultados obtidos, especialmente no modelo LSTM. A automatização permitiu explorar de forma mais eficiente o espaço de configurações, ajustando parâmetros como número de unidades ocultas, taxa de aprendizado e número de épocas. Esse processo evitou abordagens exaustivas como busca em grade e garantiu uma convergência mais rápida e eficaz dos modelos. O ganho de desempenho observado nas métricas reforça a importância da aplicação de técnicas avançadas de ajuste automático, sobretudo em modelos com maior número de parâmetros, como as redes neurais profundas.

V. CONCLUSÃO

Este trabalho avaliou o desempenho de uma rede LSTM otimizada via AutoML para a predição horária da geração de energia eólica, comparando-a com os modelos FTDNN e XGBoost. Os resultados indicaram que a LSTM apresentou melhor desempenho em todas as métricas, destacando-se pela capacidade de capturar oscilações bruscas na série temporal, como picos de geração e variações repentinas.

A aplicação do AutoML contribuiu significativamente para a escolha eficiente de hiperparâmetros, potencializando os resultados sem a necessidade de ajustes manuais. Essa combinação de LSTM e otimização automática mostrou-se promissora para séries com alta variabilidade, onde modelos tradicionais tendem a apresentar limitações.

Como trabalho futuro, propõe-se o uso de avaliação de risco operacional baseada em predições neurais, integrando a saída da LSTM com indicadores de confiabilidade para auxiliar decisões em tempo real em sistemas energéticos. A ideia é transformar as predições em escalas de risco (ex.: baixo, moderado, alto) para auxiliar o despacho de energia, manutenção

preventiva ou acionamento de fontes complementares, criando um elo entre a inteligência artificial e a operação estratégica do sistema.

REFERÊNCIAS

- S. A. Solarin, "An environmental impact assessment of fossil fuel subsidies in emerging and developing economies," *Environmental Impact Assessment Review*, vol. 85, p. 106443, 2020.
- S. A. Qadir, H. Al-Motairi, F. Tahir, and L. Al-Fagih, "Incentives and strategies for financing the renewable energy transition: A review," *Energy Reports*, vol. 7, pp. 3590–3606, 2021.
- J. A. M. Orsini, "Inct climate change phase 2 (inct mc phase 2)," 2018.
- A. C. Martins, M. d. C. Pereira, and R. Pasqualino, "Renewable electricity transition: A case for evaluating infrastructure investments through real options analysis in brazil," *Sustainability*, vol. 15, no. 13, p. 10495, 2023.
- GWEC, *Global Wind Report 2023*, 2023.
- ABBEólica, *Boletim de Dados Abr. 2023*, 2023.
- M. S. Lordêlo and Â. M. O. Sant'Anna, "Abordagens de aprendizado de máquina aplicadas à geração de energia eólica em uma estação na bahia," *International Journal of Scientific Management and Tourism*, vol. 10, no. 2, pp. e839–e839, 2024.
- J. C. Principe, N. R. Euliano, and N. R. Lefebvre, *Neural Adaptive Systems: Fundamentals Through Simulations*. John Wiley and Sons, 2000.
- T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.
- A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics*, vol. 7, p. 21, 2013.
- S. Hochreiter and J. Schmidhuber, "Lstm can solve hard long time lag problems," *Advances in neural information processing systems*, vol. 9, 1997.
- G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artificial Intelligence Review*, vol. 53, no. 8, pp. 5929–5955, 2020.
- R. C. Staudemeyer and E. R. Morris, "Understanding lstm—a tutorial into long short-term memory recurrent neural networks," *arXiv preprint arXiv:1909.09586*, 2019.
- L. Wang, X. Xu, L. Zhang, R. Qiao, M. Wu, Z. Wang, S. Zhang, J. Liang, Z. Zhang, Z. Zhang *et al.*, "Epitaxial growth of a 100-square-centimetre single-crystal hexagonal boron nitride monolayer on copper," *Nature*, vol. 570, no. 7759, pp. 91–95, 2019.
- S.-H. Noh, "Analysis of gradient vanishing of rnns and performance comparison," *Information*, vol. 12, no. 11, p. 442, 2021.
- S. Haykin, *Redes neurais: princípios e prática*. Bookman Editora, 2001.
- F. Hutter, L. Kotthoff, and J. Vanschoren, "Automated machine learning: methods, systems, challenges," *Automated Machine Learning: Methods, Systems, Challenges*, pp. 1–18, 2019.
- M. Feurer, A. Klein, K. Eggenberger, J. T. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," in *Advances in Neural Information Processing Systems*, vol. 28, 2015, pp. 2962–2970.
- T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2019, pp. 2623–2631.
- M. Ali, "hyoptxg: Optuna hyper-parameter optimization framework for predicting cardiovascular disease using xgboost," *Artificial Intelligence in Medicine*, 2024, disponível em ScienceDirect.
- M. Jafari, "Advanced hyperparameter optimization of deep learning models for wind power prediction," *Renewable Energy*, 2024, disponível em ScienceDirect.
- X. Chen, "Mapping mangrove species using multi-scale variable selection and optuna-based machine learning," *Ecological Informatics*, 2024, disponível em ScienceDirect.
- Y. Li, "Generating a daily xco2 concentration dataset in sichuan province using multi-source satellite data and stacking ensemble learning," *Remote Sensing*, 2024, disponível em ResearchGate. [Online]. Available: <https://www.researchgate.net/publication/383888806>
- M. P. Curtarelli, E. Alcântara, C. D. Rennó, A. T. Assireu, M. P. Bonnet, and J. L. Stech, "Modelling the surface circulation and thermal structure of a tropical reservoir using three-dimensional hydrodynamic lake model and remote-sensing data," *Water and environment journal*, vol. 28, no. 4, pp. 516–525, 2014.
- P. R. B. Lustosa, V. M. R. Ponte, W. R. Dominas, and C. Corrar, LJ e Theophilos, "Pesquisa operacional para decisão em contabilidade e administração," *São Paulo. Atlas*, 2004.
- G. Elliott, T. J. Rothenberg, and J. H. Stock, "Efficient tests for an autoregressive unit root," 1992.
- A. Sorjamaa, J. H. N. Reyhani, Y. Ji, and A. Lendasse, "Methodology for long-term prediction of time series," *Neurocomputing*, vol. 70, no. 16–18, pp. 2861–2869, 2007.
- J. Kingma, Diederik P e Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- R. Seidel, "Predição da temperatura do ferro-gusa em um alto-forno utilizando redes neurais lstm," 2022.