

UNIVERSIDADE FEDERAL DE PELOTAS
Instituto de Física e Matemática
Programa de Pós-Graduação em Modelagem Matemática



Dissertação

**PREVISÕES HORÁRIAS DE TEMPERATURA DO AR VIA LSTM: UMA
ABORDAGEM BASEADA EM APRENDIZADO PROFUNDO**

Milena Dias de Oliveira

Pelotas, 2026

Milena Dias de Oliveira

**PREVISÕES HORÁRIAS DE TEMPERATURA DO AR VIA LSTM: UMA
ABORDAGEM BASEADA EM APRENDIZADO PROFUNDO**

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Matemática do Instituto de Física e Matemática da Universidade Federal de Pelotas, como requisito parcial à obtenção do título de Mestre em Modelagem Matemática.

Orientador: Prof. Dr. Fabrício Pereira Härter
Coorientador: Prof. Dr. Elcio Hideiti Shiguemori

Pelotas, 2026

Universidade Federal de Pelotas / Sistema de Bibliotecas
Catalogação da Publicação

O48p Oliveira, Milena Dias de

Previsões horárias de temperatura do ar via LSTM [recurso eletrônico] : uma abordagem baseada em aprendizado profundo / Milena Dias de Oliveira ; Fabrício Pereira Härter, orientador ; Elcio Hideiti Shiguemori, coorientador. — Pelotas, 2026.

63 f.

Dissertação (Mestrado) — Programa de Pós-Graduação em Modelagem Matemática, Instituto de Física e Matemática, Universidade Federal de Pelotas, 2026.

1. Redes neurais artificiais. 2. Modelo de previsão de temperatura do Ar. 3. LSTM. 4. Inteligência artificial. I. Härter, Fabrício Pereira, orient. II. Shiguemori, Elcio Hideiti, coorient. III. Título.

CDD 551.63

**PREVISÕES HORÁRIAS DE TEMPERATURA DO AR VIA LSTM: UMA
ABORDAGEM BASEADA EM APRENDIZADO PROFUNDO**

por

Milena Dias de Oliveira

Dissertação submetida ao Programa de Pós-Graduação em Modelagem Matemática, PPGMMat, do Instituto de Física e Matemática da Universidade Federal de Pelotas, como parte dos requisitos necessários para a obtenção do Título de

Mestre em Modelagem Matemática

Banca Examinadora:

Prof. Dr. Fabrício Pereira Härter – Orientador (UFPel)

Prof. Dr. Elcio Hideiti Shiguemori – Co-Orientador (INPE)

Prof. Dr. Rose Ane Pereira de Freitas (UFPel)

Prof. Dr. Regis Sperotto de Quadros (UFPel)

Profa. Dra. Julio Renato Quevedo Marques (UFPel)

Pelotas, 23 de fevereiro de 2026

AGRADECIMENTOS

Agradeço a Deus, que me fortalece todos os dias.

À minha família que são minha base, principalmente meus pais, pelo total apoio e incentivo a prosseguir nos estudos, mesmo sendo necessário, com muita dor, morar longe deles e perder as datas comemorativas e os momentos costumeiros juntos.

Aos meus amigos Varlei Barbosa, Aline Timm, Welison de Oliveira e Lázaro Nunes, que estão na mesma caminhada acadêmica e a tornam mais leve, com dicas, desabafos, apoio e cafés compartilhados.

Agradeço ao meu orientador Fabrício Härter por me acolher e acreditar no meu potencial. Agradeço ao professor Júlio Renato Quevedo Marques da faculdade de meteorologia pela ajuda com uma etapa importante dessa pesquisa.

Agradeço a oportunidade me dada pelo programa de pós-graduação em Modelagem Matemática da UFPEL de aprender novos conteúdos e viver experiências que me proporcionaram um amadurecimento acadêmico e ampliaram minha visão de pesquisadora. Agradeço também à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela bolsa.

Enfim, agradeço a mim mesma por não desistir, mesmo quando essa parecia ser a melhor opção.

A viagem de mil quilômetros começa com um passo.
— LAO TZU

RESUMO

OLIVEIRA, Milena Dias de. **PREVISÕES HORÁRIAS DE TEMPERATURA DO AR VIA LSTM: UMA ABORDAGEM BASEADA EM APRENDIZADO PROFUNDO**. Orientador: Fabrício Pereira Härter. 2026. 63 f. Dissertação (Mestrado em Modelagem Matemática) – Instituto de Física e Matemática, Universidade Federal de Pelotas, Pelotas, 2026.

Considerando a importância das previsões de temperatura do ar no monitoramento climático e no cotidiano das pessoas, por influenciar aspectos sociais e econômicos, essa pesquisa teve como principal objetivo realizar previsões horárias de temperatura do ar em diferentes escalas de tempo utilizando um modelo baseado em aprendizado profundo via rede neural artificial recorrente *Long Short-Term Memory* (LSTM). Para isso, foi desenvolvido um modelo computacional na linguagem de programação Python composto por duas camadas da rede LSTM, tendo como variáveis de entrada séries temporais de dados horários de temperatura do ar, pressão, temperatura do ponto de orvalho e velocidade do vento, referentes aos anos de 2001 a 2024 da cidade de Porto Alegre/RS. O modelo foi treinado em diferentes cenários, sendo um deles com o conjunto de dados do INMET e outro com os dados de reanálise do ERA5, considerando tanto o conjunto completo quanto separados em subconjuntos de dados sazonais. Após cada um desses processos de treinamento, foram geradas previsões horárias de temperatura do ar para o horizonte de 6h a 30 dias. Os resultados obtidos mostram previsões de temperaturas horárias competitivas com previsões de modelos físicos/dinâmicos na escala de Nowcasting (até 6 horas), superiores na escala sinótica até 10 dias (maior do que 6 horas e inferior a 15 dias) e competitiva na escala subsazonal (15 a 30 dias). A LSTM treinada com dados de reanálise apresentou melhores resultados do que treinada com estações automáticas do INMET. O modelo neural apresentou melhores resultados quando treinado com o conjunto de treinamento total, em comparação as redes treinadas por estações do ano. A LSTM treinada com 4 preditores (pressão à superfície, temperatura do ar à 2 m da superfície, temperatura do ponto de orvalho e vento à superfície) resultou em previsões de temperatura horária mais precisas do que a rede treinada apenas com temperatura do ar.

Palavras-chave: Redes Neurais Artificiais; Modelo de Previsão de Temperatura do Ar; LSTM; Inteligência Artificial.

ABSTRACT

OLIVEIRA, Milena Dias de. **Hourly Air Temperature Forecasts via LSTM: A Deep Learning-Based Approach**. Advisor: Fabrício Pereira Härter. 2026. 63 f. Dissertation (Masters in Mathematical Modeling) – Institute of Physics and Mathematics, Federal University of Pelotas, Pelotas, 2026.

Given the importance of air temperature forecasts for climate monitoring and daily activities, due to their influence on social and economic processes, this study aimed to develop hourly air temperature forecasts at different temporal scales using a deep learning approach based on a Long Short-Term Memory (LSTM) recurrent neural network. A computational model was implemented in the Python programming language, consisting of two LSTM layers, with input variables including hourly time series of air temperature, surface pressure, dew point temperature, and wind speed. The dataset covers the period from 2001 to 2024 for the city of Porto Alegre, Rio Grande do Sul, Brazil. The model was trained under multiple scenarios: one using observational data from the Brazilian National Institute of Meteorology (INMET) and another using ERA5 reanalysis data, considering both the complete dataset and seasonally segmented subsets. After each training procedure, hourly air temperature forecasts were generated for a 30-day leading time. The results indicate that the proposed LSTM model produces hourly temperature forecasts that are competitive with physical and dynamical models at the nowcasting scale (up to 6 hours), outperform them at the synoptic scale up to 10 days (greater than 6 hours and less than 15 days), and remaining competitive at the subseasonal scale (15 to 30 days). The LSTM model trained with reanalysis data exhibited superior performance compared to the model trained with INMET automatic weather station data. Additionally, the neural network achieved better results when trained using the complete dataset rather than seasonally partitioned datasets. Finally, the LSTM trained with four predictors (surface pressure, 2 m air temperature, dew point temperature, and surface wind) yielded more accurate hourly air temperature forecasts than the model trained using air temperature alone.

Keywords: Artificial Neural Networks; Air Temperature Forecasting Model; LSTM; Artificial intelligence.

LISTA DE FIGURAS

Figura 1	Loop interno de uma RNR. Fonte: Olah et al. (2015)	19
Figura 2	Arquitetura da rede LSTM. Fonte: Silva (2021)	20
Figura 3	Portão de esquecimento da LSTM. Fonte: Olah et al. (2015)	20
Figura 4	Portão de Entrada da LSTM. Fonte: Olah et al. (2015)	21
Figura 5	Portão de saída da LSTM. Fonte: Olah et al. (2015)	22
Figura 6	Fluxograma metodológico. Fonte: Elaboração própria	24
Figura 7	Ponto de localização e registro fotográfico da Estação meteorológica do INMET em Porto Alegre/RS. Fonte: Adaptado de Google (2025)	25
Figura 8	Gráfico com os valores do RMSE e o seu limite aceitável para cada horizonte de previsão.	33
Figura 9	Gráfico de previsão horária de temperatura do ar no período de 24 horas.	35
Figura 10	Gráfico de previsão horária de temperatura do ar no período de 7 dias.	35
Figura 11	Gráfico de previsão horária de temperatura do ar no período de 30 dias.	36
Figura 12	Desempenho do modelo ao realizar previsões de temperatura do ar para diferentes horizontes de previsão ao ser treinado com dados do ERA5	37
Figura 13	Gráfico de previsão horária de temperatura do ar no período de 24 horas realizada ao modelo ser treinado com dados do ERA5	38
Figura 14	Gráfico de previsão horária de temperatura do ar no período de 7 dias gerada pelo modelo ao ser treinado com dados do ERA5	39
Figura 15	Gráfico de previsão horária de temperatura do ar no período de 30 dias gerada pelo modelo ao ser treinado com dados do ERA5	39
Figura 16	Valores do RMSE das previsões geradas com o modelo LSTM treinado apenas com uma variável e ao ser treinado com várias variáveis	41
Figura 17	Gráfico de comparação: previsões horárias de temperatura do ar no período de 24h geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas.	42
Figura 18	Gráfico de comparação: previsões horárias de temperatura do ar no período de 7 dias geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas	42
Figura 19	Gráfico de comparação: previsões horárias de temperatura do ar no período de 30 dias geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas	43

Figura 20	Gráfico com os valores do RMSE das previsões referentes aos horizontes de previsão de 24h, 7 e 30 dias obtidas após o modelo ser treinado com os quatro conjuntos de dados sazonais	44
Figura 21	Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 24 horas	46
Figura 22	Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 7 dias	46
Figura 23	Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 30 dias	47

LISTA DE TABELAS

Tabela 1	Valores das Métricas de Avaliação para as previsões realizadas durante o mês de dezembro de 2024.	32
Tabela 2	Valores das Métricas de Avaliação para as previsões realizadas durante o mês de dezembro de 2024 com o modelo treinado com dados do ERA5.	37

LISTA DE ABREVIATURAS E SIGLAS

LSTM	Long Short-Term Memory
INMET	Instituto Nacional de Meteorologia
ERA5	Reanálise do ECMWF Quinta Geração
RNA	Rede Neural Artificial
RNR	Rede Neural Recorrente
IA	Inteligência Artificial
VS-Code	Visual Studio Code
MSE	Erro Quadrático Médio
RMSE	Raiz do Erro Médio Quadrático
MAE	Erro Médio Absoluto
ECMWF	European Centre for Medium-Range Weather Forecasts
WRF	Weather Research and Forecasting Model
GFS	Sistema Global de Previsão
MetNet	Google Neural Weather Model
DeepAR	Deep Autoregressive
IFS	Sistema Integrado de Previsão
FourCastNet	Fourier Forecasting Neural Network
GNN	Graph neural networks

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Objetivos da Pesquisa	15
1.1.1	Objetivo geral	15
1.1.2	Objetivos específicos	16
1.2	Justificativa	16
1.3	Organização do Trabalho	17
2	REVISÃO BIBLIOGRÁFICA	18
2.1	Previsão de Temperatura do Ar por Redes Neurais Artificiais	18
2.1.1	A rede <i>Long Short-Term Memory</i> (LSTM)	19
2.1.2	A LSTM aplicada na previsão de temperatura	22
3	METODOLOGIA	24
3.1	Processamento dos Dados	24
3.2	Definição de Alguns Termos Importantes sobre o Modelo	26
3.3	Arquitetura do Modelo	27
3.4	Métricas de Avaliação	29
3.5	Procedimentos para Gerar as Previsões	30
4	RESULTADOS	32
4.1	Modelo Treinado com Dados do INMET	32
4.2	Modelo Treinado com Dados do ERA5	36
4.2.1	Treinamento com Várias Variáveis Meteorológicas	37
4.2.2	Comparação do Modelo treinado com uma Variável e com várias Variáveis Meteorológicas	40
4.3	Modelo Treinado com Dados Sazonais	43
4.4	Comparação do modelo LSTM com o ERA5	45
5	CONCLUSÕES	48
	REFERÊNCIAS	50
	APÊNDICE A ALGORITMO UTILIZADO PARA TREINAMENTO DO MODELO	56
	APÊNDICE B ALGORITMO UTILIZADO PARA GERAR AS PREVISÕES	59

1 INTRODUÇÃO

A previsão de temperatura do ar é um dos principais fatores no planejamento climático e socioambiental, pois influencia diretamente aspectos sociais e econômicos (Castro et al., jul. 2020). Sua relevância se manifesta tanto no cotidiano das pessoas, que adquiriu o hábito de verificar a previsão do tempo antes de planejar atividades e eventos, como plantações, passeios, viagens e qual vestimenta adequada usar, quanto em escala global, especialmente no contexto do desenvolvimento sustentável, na dinâmica de interação solo-atmosfera, proteção dos ecossistemas, gerenciamento de recursos hídricos e na emissão de alertas de desastres climáticos (Hou et al., 2022).

Nesse sentido, essas previsões são fundamentais nas medidas preventivas de setores essenciais para a sociedade como a agricultura e pecuária, além de exercer impacto crucial na medição do efeito estufa, estimativas de radiação solar e poluição do ar (Dhital et al., 2024). Além disso, desempenha um papel relevante no cuidado com a saúde, ao possibilitar alertas destinados a pessoas vulneráveis às variações térmicas, que podem elevar o risco de problemas como ataques cardíacos (Rowland et al., 2020). Dessa forma, torna-se indispensável o conhecimento antecipado do comportamento da temperatura do ar ao longo do tempo, a fim de identificar a ocorrência de possíveis anomalias (Santos; Mattos; Krebs, 2002).

Para gerar essas previsões faz-se uso da modelagem matemática e computacional onde é necessário que os modelos utilizados estejam coerentes com os fatores que envolvem e descrevem a temperatura atmosférica. Contudo, os cálculos necessários são complexos e demorados, o que é inviável de serem resolvidos manualmente, por isso se faz necessário o uso de supercomputadores. Mesmo assim, os modelos numéricos tradicionais utilizados nessas previsões exigem uma elevada carga computacional e uma execução demorada para gerar as previsões (Artaxo et al., 2024).

Por esse motivo a ciência busca constantemente por novos e mais precisos métodos para a previsão do tempo com menor custo computacional. Algumas soluções propostas indicam o uso de modelos de aprendizado profundo, ou simplesmente modelos de IA, que fazem uso de Redes Neurais Artificiais (RNAs) (Artaxo et al., 2024).

Esses modelos têm ganhado destaque nas previsões meteorológicas devido a sua alta capacidade de lidar com uma grande quantidade de dados, identificar padrões complexos, como os presentes nas séries temporais (Dhital et al., 2024; Pham et al., 2020) e assim realizar simulações futuras.

Os primeiros modelos de aprendizado profundo surgiram na década de 1990, e as pesquisas extensivas sobre o uso desses modelos na previsão do tempo em escala local e de curto prazo começaram a ser realizadas em meados da década de 2010, porém as pesquisas nesta área do conhecimento têm aumentado nos últimos anos (Olivetti; Messori, março, 2024). No artigo de Schultz et al. (abril, 2021) os autores observaram que o aprendizado profundo na área da meteorologia necessita de avanços para poder competir com as previsões com modelos de equações primitivas.

Desde então, esta área de pesquisa tem atraído vários pesquisadores, e em alguns trabalhos como os de Pathak et al. (2022); Bi et al. (2023); Keisler (2022); Lam et al. (2023); Chen et al. (2023a); Nguyen et al. (2023); Chen et al. (2023b), os autores afirmam terem desenvolvido modelos de aprendizado profundo capazes de realizar previsões de variáveis meteorológicas de curto e médio prazo mais precisas que as do Centro Europeu de Previsões Meteorológicas de Médio Prazo (ECMWF), que são reconhecidamente previsões de alta precisão (Olivetti; Messori, março, 2024).

O diferencial no aprendizado profundo, são as RNA com arquitetura recorrente (RNR), inspiradas na inteligência biológica, possuindo memória capaz de guardar um grande volume de dados e memorizar padrões (Neto et al., 2020). Dentre elas, destacam-se a rede recorrente *Long Short-Term Memory* (LSTM), promissora na previsão do tempo (Chollet, 2021; Dhital et al., 2024), que surgiu como proposta de solucionar um problema comum nos modelos de RNR denominado como o desaparecimento ou explosão do gradiente (Hochreiter; Schmidhuber, 1997), que acontece quando o gradiente da função custo tende a zero ou quando seu valor cresce demasiadamente, fazendo com que o mínimo global na superfície de erros não seja alcançado.

1.1 Objetivos da Pesquisa

Além de ser importante para as atividades humanas e no monitoramento climático, as previsões de temperatura do ar também são usadas como dados de entrada para outros modelos, como os hidrológicos e os agrícolas. Nesse contexto, essa pesquisa tem os seguintes objetivos:

1.1.1 Objetivo geral

- Realizar previsões horárias pontuais de temperatura do ar em diferentes escalas temporais (até 30 dias) utilizando um modelo de aprendizado profundo baseado na rede neural artificial recorrente LSTM.

1.1.2 Objetivos específicos

- Processar dados históricos horários de temperatura do Ar da cidade de Porto Alegre no Rio Grande do Sul.
- Desenvolver um algoritmo na linguagem de programação Python utilizando a rede neural recorrente LSTM para gerar as previsões horárias de temperatura do ar.
- Treinar o modelo neural desenvolvido com dados observacionais do INMET e com dados de reanálise do ERA5 e comparar os resultados obtidos.
- Treinar o modelo com subconjuntos de dados sazonais (Estações do ano) do ERA5 e comparar os resultados.
- Treinar o modelo utilizando um conjunto de dados com apenas uma variável: a temperatura do ar, e comparar os resultados ao ser treinado com um conjunto de quatro variáveis meteorológicas: temperatura do ar, pressão, temperatura no ponto de orvalho e velocidade do vento.
- Avaliar o desempenho do modelo neural LSTM e a qualidade das previsões utilizando métricas de avaliação consolidadas na literatura científica.
- Comparar a qualidade das previsões geradas pelo modelo neural LSTM, treinado com dados do INMET, com a reanálise do ERA5.

1.2 Justificativa

A motivação para essa pesquisa surgiu da crescente preocupação com as mudanças climáticas e do engajamento em compreender o potencial das ferramentas tecnológicas que estão impactando e transformando o mundo, como é o caso da Inteligência Artificial (IA) com as redes neurais artificiais, cuja aplicação é promissora em áreas estratégicas como a previsão de temperatura do ar, que é uma variável essencial no monitoramento climático.

No entanto, apesar dos avanços internacionais nessa área, ainda existe uma carência de pesquisas, principalmente no Brasil, que explorem de forma sistemática o uso desses modelos na previsão de variáveis meteorológicas. Essa lacuna é ainda mais evidente quando se considera o uso específico da rede neural recorrente LSTM.

Embora a LSTM seja amplamente reconhecida por sua capacidade de capturar dependências temporais de curto e longo prazo, sendo particularmente promissora para previsões de múltiplos passos à frente (Yunpeng; Di; Junpeng; Yong, 2017), são relativamente raros os estudos que investigam sua aplicação na previsão horária da temperatura do ar, especialmente em múltiplas escalas temporais que variam até 30

dias. A maioria dos estudos foca em previsões de curto prazo, o que limita a análise da capacidade de generalização desse modelo ao longo do tempo.

Dessa forma, identifica-se uma lacuna relevante na literatura quanto ao desenvolvimento e à avaliação de modelos baseados na rede LSTM capazes de realizar previsões horárias pontuais da temperatura do ar em diferentes escalas temporais, especialmente em regiões com alta variabilidade climática e menor disponibilidade de estudos, como o Brasil. Logo, esta pesquisa contribui para o avanço do conhecimento científico nas áreas de previsão do tempo, modelagem matemática e computacional.

1.3 Organização do Trabalho

Seguido da introdução, este trabalho está organizado da seguinte forma:

- Capítulo 2: tem-se o referencial teórico, descrevendo sobre as Redes Neurais artificiais, tendo uma parte introdutória sobre a IA e seus subcampos o aprendizado de máquina e o aprendizado profundo. Também apresenta um subtópico detalhado sobre a LSTM. Após, tem-se a seção apresentando trabalhos relevantes e o estado da arte em modelos de aprendizado profundo no uso de previsões de variáveis meteorológicas.
- Capítulo 3: é apresentada a Metodologia utilizada na pesquisa, dando ênfase aos instrumentos e técnicas utilizadas para a preparação do modelo e realização das previsões.
- Capítulo 4: É onde se tem os resultados obtidos, apresentados em forma de gráficos e tabelas, e descrito as suas interpretações.
- Capítulo 5: Tem-se as considerações finais, mostrando a conclusão sobre o desempenho do modelo e os resultados alcançados, além de um direcionamento para uma possível continuação desta pesquisa.

2 REVISÃO BIBLIOGRÁFICA

2.1 Previsão de Temperatura do Ar por Redes Neurais Artificiais

A temperatura do ar é um dos elementos do tempo atmosférico mais discutido e pode ser definida, tomando como base o grau de calor de um corpo, como a condição que determina o fluxo de calor que passa de uma substância à outra (Ayoade, 1996). Pode ser expressa nas escalas Fahrenheit, Celsius e Kelvin, sendo a escala Celsius a mais utilizada. Ela é uma variável importante na medição de balanço de energia, hidrológico, o efeito estufa e a estimativa da radiação solar total e a poluição do ar (Dhital et al., 2024).

Para realizar essas previsões são, tradicionalmente, utilizados modelos numéricos baseados em física. No entanto, fatores que afetam as mudanças de temperatura são extremamente complexos e não lineares, tornando difícil para os métodos estatísticos capturarem essas mudanças ao prever longas séries temporais, levando a imprecisão nas previsões de temperatura horárias e diárias (Hou et al., 2022; Ramesh; Anitha, 2014).

É nesse cenário que o Aprendizado profundo com as RNA começam a dominar, devido à sua capacidade inata de modelar dados complexos e não-lineares sem a necessidade de equações físicas pré-definidas (Haykin, 2009). Pesquisas como os de Schizas et al. (1991), Kaur; Sharma; Agrawal (2011), Chollet (2021), Haque; Tabassum; Hossain (2021), Hou et al. (2022), Bi et al. (2023), Díaz-ramíre et al. (2024), Artaxo et al. (2024), (Lam et al., 2023) e (Pathak et al., 2022), mostram que modelos baseados em aprendizado profundo com redes neurais artificiais tem potencial significativo para complementar e até superar as técnicas clássicas nas previsões de variáveis meteorológicas, tais como a temperatura do ar (Artaxo et al., 2024).

As redes neurais artificiais (RNAs) foram introduzidas por McCulloch; Pitts (1943) com base nos neurônios biológicos que são capazes de memorizar padrões. A principal característica das RNAs é a sua habilidade de aprendizado a partir do ambiente e consequentemente o melhoramento em seu desempenho (Castro et al., jul. 2020).

Entre as arquiteturas de redes neurais utilizadas em previsão meteorológica,

destacam-se as redes com arquitetura recorrente. Essas redes possuem um *loop* interno entre os dados de entrada e saída (Figura 1), que mantém o modelo constantemente atualizado ao processar novas informações e interagir com as anteriores.

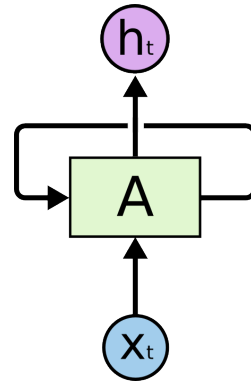


Figura 1 – Loop interno de uma RNR. Fonte: Olah et al. (2015)

Entretanto, redes recorrentes tradicionais apresentam limitações durante o processo de treinamento, principalmente devido ao problema conhecido como desaparecimento ou explosão do gradiente, que ocorre quando, durante o treino das redes profundas, os gradientes se tornam muito pequenos, dificultando a aprendizagem nas camadas iniciais ou se tornam demasiado grandes, causando instabilidade e dificultando a convergência do modelo impedindo o seu aprendizagem de dependências temporais de longo prazo (Bengio; Simard; Frasconi, 1994).

2.1.1 A rede *Long Short-Term Memory* (LSTM)

Como proposta de solucionar esse problema, Hochreiter; Schmidhuber (1997) propuseram a rede LSTM. Essa rede é conhecida por sua capacidade de processar longas sequências de dados, entender a relação entre as variáveis envolvidas e descrever sistemas não-lineares e fenômenos complexos pelo método de treinamento (Castro et al., jul. 2020), obtendo, assim, a capacidade de realizar simulações futuras. Sua arquitetura (Figura 2) é formada por três portões (ou *gates*): o de esquecimento, de entrada e o de saída, que controlam o fluxo das informações de cada neurônio (Hochreiter; Schmidhuber, 1997).

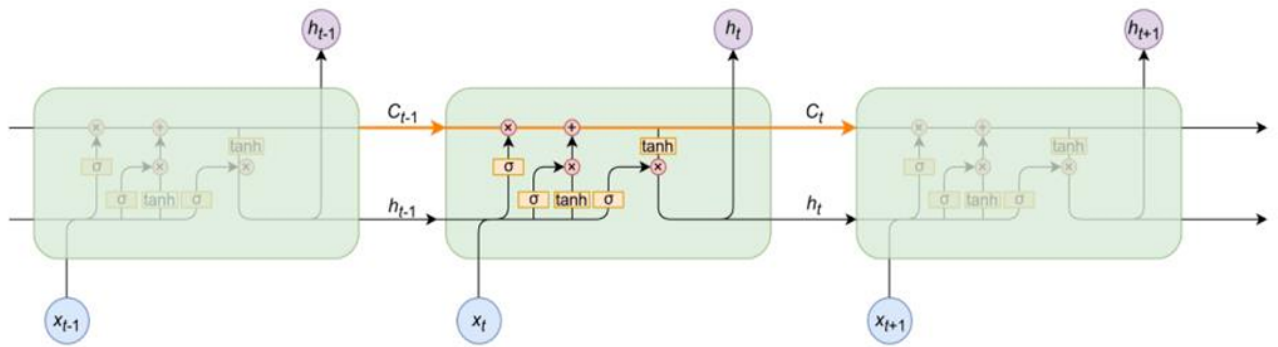


Figura 2 – Arquitetura da rede LSTM. Fonte: Silva (2021)

- **Portão de esquecimento** (*forget gate*), ilustrado na Figura 3: decide quais informações são relevantes e devem permanecer e quais devem ser descartadas ou esquecidas (Hochreiter; Schmidhuber, 1997).

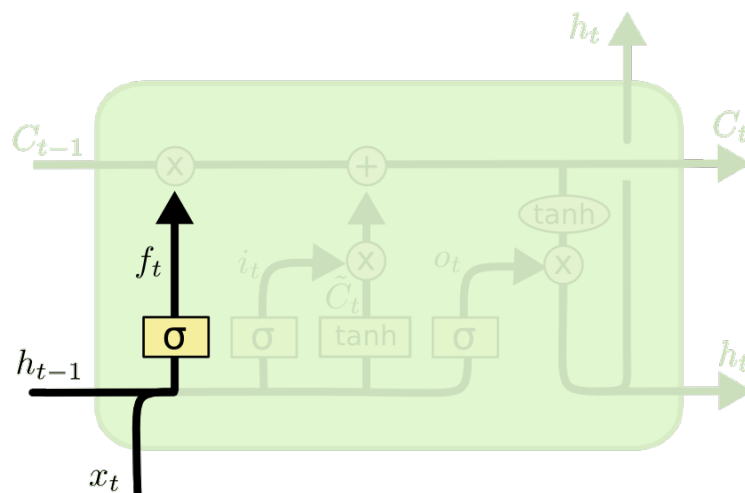


Figura 3 – Portão de esquecimento da LSTM. Fonte: Olah et al. (2015)

O portão de esquecimento é formado por uma camada sigmoide representada pela equação 1, que resulta em valores entre 0 e 1 para cada estado da célula (Silva, 2021). Se o valor for 0 significa que não pode deixar passar nenhuma informação e o estado interno é limpo. Se o valor for 1 significa que tudo pode passar e o estado interno se mantém intacto.

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

onde σ é a função sigmoide, W_f (peso) é uma matriz de parâmetros que recebe $h_{(t-1)}$ (estado anterior) e x_t (entrada no tempo), e b_f é um valor de correção.

- **Portão de entrada** (*input gate*), representado pela figura 4: decide quais informações são importantes para serem adicionadas à memória de curto prazo

e quais não são relevantes e irão ser descartadas. As células armazenam as informações relevantes no estado da célula (Hochreiter; Schmidhuber, 1997).

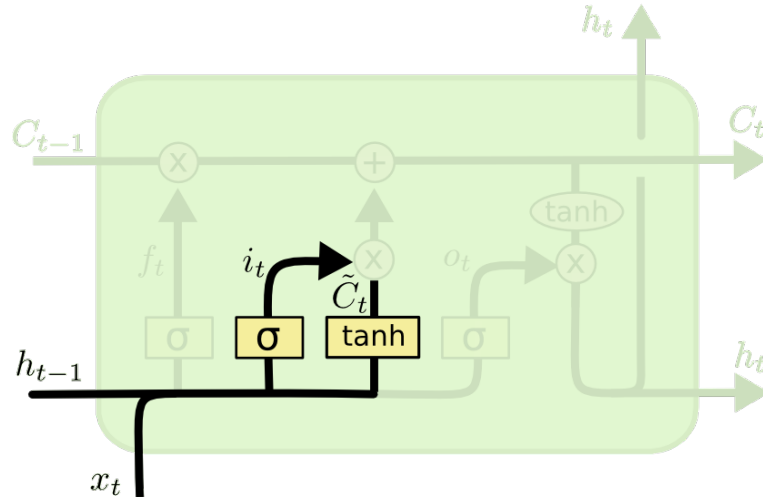


Figura 4 – Portão de Entrada da LSTM. Fonte: Olah et al. (2015)

O portão de entrada é formado por duas camadas sigmoide, representadas pelas equações:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

onde W_i e W_c (pesos) são matrizes de parâmetros que recebem $h_{(t-1)}$ e x_t , e b_i e b_c são valores de correlação.

Por meio da camada sigmoide, representada pela equação 2, os cálculos iniciais são realizados e determinam quais informações serão atualizadas. Em seguida, a camada sigmoide, representada pela equação 3, criará um estado da célula \tilde{C}_t gerando novos valores candidatos utilizando a função \tanh (tangente hiperbólica), que converte os valores para ficarem entre -1 e 1, evitando o erro de explosão ou desaparecimento do gradiente.

A combinação do resultado dos cálculos das equações 2 e 3 geram uma nova entrada no estado da célula representado pela equação 4:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (4)$$

Observe que o antigo estado da célula $C_{(t-1)}$ é multiplicado por f_t , isso faz com que esse estado seja preservado em todo o processo. Após são adicionados os novos valores candidatos ($i_t \cdot \tilde{C}_t$).

- **Portão de saída (output gate)**, figura 5: decide o que vai ser gerado pela saída com base no estado da célula, isto é, “quais partes do estado da célula são

importantes no instante atual para gerar a saída (*output*) da célula” (Silva, 2021, p.44) e quais informações serão passadas para o próximo tempo. A saída é comparada com a saída esperada e o erro é calculado usando alguma função de erro.

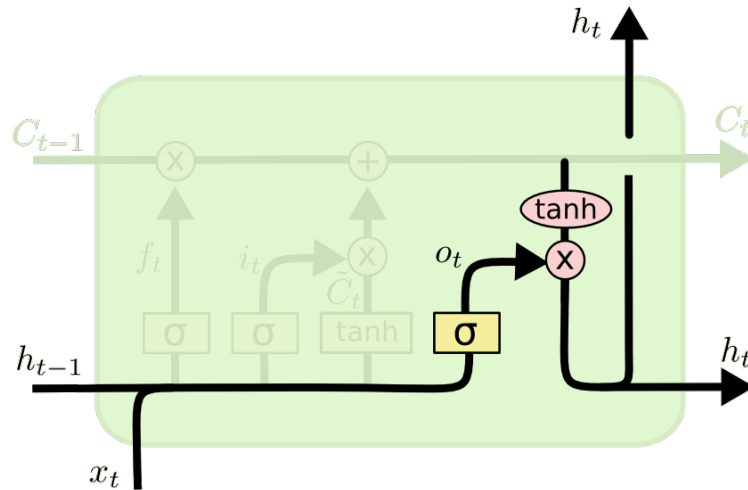


Figura 5 – Portão de saída da LSTM. Fonte: Olah et al. (2015)

Este portão é formado por duas camadas sigmoides, representados pelas equações 5 e 6:

$$O_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = O_t \cdot \tanh(C_t) \quad (6)$$

Por meio da camada sigmoide, representada pela equação 5, é realizado o primeiro cálculo, o qual decide que parte do estado da célula passará pela saída. Em seguida, usando a equação 6, o segundo cálculo é realizado para gerar as saídas. Multiplicando entre si os resultados obtidos nos cálculos da equação 5 e 6, é gerada a saída de uma célula (Silva, 2021).

2.1.2 A LSTM aplicada na previsão de temperatura

Por causa da sua estrutura baseada em memória controlada, as redes LSTM apresentam grande capacidade de aprendizado em séries temporais complexas, tornando-se amplamente utilizadas na previsão de variáveis meteorológicas, incluindo a temperatura do ar.

Diversos estudos recentes têm explorado o uso de LSTM e outras arquiteturas de aprendizado profundo para melhorar a precisão das previsões meteorológicas. Hou et al. (2022), por exemplo, propuseram um modelo híbrido combinando redes convolucionais (CNN) e LSTM para previsão horária da temperatura do ar. Nesse modelo, a CNN é utilizada para extrair características relevantes dos dados, enquanto a LSTM

modela as dependências temporais da série. Os resultados demonstraram que a abordagem híbrida apresentou alta precisão nas previsões.

Nos últimos anos, avanços no campo do aprendizado profundo também resultaram no desenvolvimento de modelos meteorológicos globais baseados em redes neurais. Entre os principais modelos destacam-se o **GraphCast**, o **Pangu-Weather** e o **FourCastNet**, conhecidos por apresentarem elevado desempenho na previsão do tempo.

O **GraphCast** é um modelo meteorológico de curto prazo utilizado pelo sistema Google DeepMind para realizar previsões contínuas de 24 horas, sendo a temperatura uma delas. Utiliza redes *graph neural networks* (GNNs) e, após treinado, consegue produzir previsões de 10 dias em menos de um minuto. Suas previsões são tão precisas quanto as de outras agências meteorológicas e superam a previsão do ECMWF em até 10 dias (Lam et al., 2023; Artaxo et al., 2024).

O modelo **Pangu-Weather**, desenvolvido pela academia chinesa Huawei, foi o primeiro sistema baseado em aprendizado profundo a superar em precisão os métodos de previsão numérico do tempo de última geração. O modelo consiste no uso de redes neurais com arquitetura 3D (3D-Transformers) e apresenta grandes vantagens em previsões de curto e médio prazo (de uma hora a uma semana) (Bi et al., 2023).

O **FourCastNet**, abreviado de *Fourier Forecasting Neural Network*, é um modelo global de previsão do tempo de curto e médio prazo que utiliza operadores neurais de Fourier adaptativos. Ele iguala, em termos de precisão, as previsões do Sistema Integrado de Previsão (IFS) do ECMWF (Pathak et al., 2022).

Além desses, outros modelos baseados em aprendizado profundo para previsão do tempo conhecidos na literatura que geram previsões com alta precisão são o **ClimaX** (Nguyen et al., 2023), o **MetNet** (Google Neural Weather Model) (Sønderby et al., 2020), e o **DeepAR** (Deep Autoregressive) (Salinas et al., 2020). Além disso, os modelos de IA também são usados para corrigem lacunas de alguns modelos baseados em física reconhecidos, tais como os do **ECMWF** (Frnda et al., 2022), o **GFS** (Sistema Global de Previsão) (Guo; Yang; Liu, 2024), e o **WRF** (*Weather Research and Forecasting Model*) (Furnari et al., 2025).

Dando ênfase aos modelos de previsão de temperatura do ar utilizando a LSTM, destaca-se o trabalho de Chollet (2021), base desta pesquisa. O autor comparou o desempenho da LSTM com um modelo convolucional unidimensional (1D-CNN) e com um método simples sem aprendizado de máquina ao realizar previsões horárias de temperatura do ar para horizonte de 24 horas. Os resultados mostraram que a LSTM apresentou melhor desempenho, obtendo menores erros médios absolutos em comparação aos demais modelos.

Partindo desse trabalho, a presente pesquisa utiliza a rede LSTM para realizar previsões da temperatura do ar em diferentes horizontes de previsão, buscando verificar se mantém o bom desempenho, principalmente para previsões de longo prazo.

3 METODOLOGIA

Esta é uma pesquisa de cunho quantitativo e de natureza aplicada, com o objetivo de desenvolver um modelo baseado em aprendizado profundo para prever a temperatura do ar horária em diferentes escalas temporais. A figura 6 apresenta o fluxograma descrevendo os passos metodológicos adotados. A estrutura da metodologia deste trabalho consiste nos seguintes tópicos:

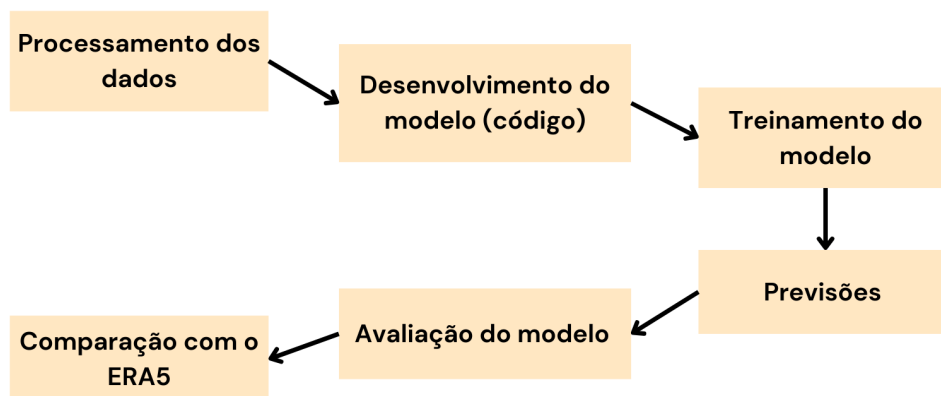


Figura 6 – Fluxograma metodológico. Fonte: Elaboração própria

3.1 Processamento dos Dados

Nessa pesquisa foram utilizados dados observados horários de temperatura do ar, pressão, temperatura no ponto de orvalho e velocidade do vento disponibilizados no site do Instituto Nacional de Meteorologia (INMET) (Instituto Nacional de Meteorologia, 2026), com registros horários entre os anos de 2001 e 2024, totalizando 24 anos ou 210.384 horas. Os dados foram organizados como séries temporais, sendo feita a concatenação de todos os valores (2001-2024) em um único documento no formato CSV

(valores separados por vírgula) contendo apenas cinco colunas, a primeira contendo a data e hora, e as demais contendo os valores das quatro variáveis meteorológicas utilizadas.

Esses dados referem-se à estação meteorológica automática localizada no Jardim Botânico em Porto Alegre no Rio Grande do Sul (Figura 7), cujas coordenadas são: (-30.05, -51.17). A Figura 7 mostra o ponto de localização dessa estação no mapa (imagem a esquerda) e um registro fotográfico dela (imagem a direita). Essa área de pesquisa foi escolhida por ter um longo registro de dados e pela facilidade em obtê-los.



Figura 7 – Ponto de localização e registro fotográfico da Estação meteorológica do INMET em Porto Alegre/RS. Fonte: Adaptado de Google (2025)

Em decorrência de os dados do INMET possuírem lacunas como dados faltantes e registros anômalos, podendo comprometer o treinamento do modelo, optou-se por utilizar também dados de reanálise do modelo global denominado ERA5 produzidos pela ECMWF (European Centre for Medium-Range Weather Forecasts, 2026). Foram consideradas as mesmas quatro variáveis meteorológicas, sendo elas a temperatura do ar a 2m da superfície, temperatura do ponto de orvalho a 2m, pressão à superfície e velocidade do vento a 10 m da superfície. Os dados do ERA5 são espacial e temporalmente contínuos, não apresentam valores faltantes e possuem elevada qualidade e consistência física (Hersbach et al., 2020; Perleberg et al., 2025).

Tanto a série temporal definida pelo conjunto de dados do INMET quanto do ERA5, passaram pelo processo de retirada da tendência. A remoção da tendência é necessária porque evita falsas relações estatísticas e ajuda a atingir estacionalidade da série, gerando uma série mais adequada para análises estatísticas e métodos de previsão (Brownlee, 2020). Dessa forma, remover a tendência favorece o aprendizado do modelo, facilitando a análise de extremos.

A tendência linear de todas as séries de dados utilizadas para treinamento do modelo foi calculada utilizando a forma geral da regressão linear, representada pela equação 7, escolhida por ser um método simples e de fácil interpretação (Wilks, 2019). Em seguida essa tendência era removida utilizando a equação 8. Todo esse processo foi realizado em um código na linguagem Python utilizando a função *statsmodels*.

$$y_t = a + bt \quad (7)$$

onde y_t é o valor estimado da variável meteorológica no tempo, t é o tempo, a é o valor inicial da série e b é o coeficiente angular que representa a tendência.

$$y_{st} = y_t - \hat{y}_t \quad (8)$$

onde y_{st} representa os dados sem tendência, y_t representa os dados no valor inicial, e \hat{y} a tendência estimada.

3.2 Definição de Alguns Termos Importantes sobre o Modelo

É importante para a compreensão do modelo entender alguns termos que o compõem, dentre eles estão:

- **Pesos:** São parâmetros ajustáveis que determinam a influência das conexões entre os neurônios em diferentes camadas.
- **Batch size:** Número de amostras usados em uma iteração do algoritmo antes de atualizar os pesos.
- **Época:** É uma passagem completa por todo o conjunto de amostras (*batch size*) no processo de treinamento do modelo. Por exemplo, se o conjunto de dados tem 64 amostras, então em uma época a rede vê todas essas amostras e em seguida realiza ajustes nos pesos com base nos erros cometidos, possibilitando o aprendizado do modelo.
- **Look back:** É usado tanto na etapa de treinamento quanto na de teste. Como as redes LSTM são recorrentes, elas são capazes de processar longas sequências de dados mantendo informações dos instantes anteriores. Na etapa de treinamento o *look back* determina o tamanho da janela de entrada, isto é, quantos valores anteriores serão mostrados em cada passo, e assim realiza uma sequência completa de eventos consecutivos. Ao processar essas sequências temporais, uma de cada vez, determinada pelo valor definido do *look back*, o modelo, então, consegue aprender padrões temporais. Na etapa de teste, o *look back* define quantas observações anteriores o modelo vai utilizar para gerar cada previsão, ou seja, quantos valores do passado serão usados como base para prever os próximos valores.
- **Overfitting:** é um sobreajuste que acontece quando o modelo aprende além do necessário com os dados de treino, incluindo padrões irrelevantes e ruídos. Isso

faz com que o modelo tenha um excelente desempenho no treinamento, mas falhe nos testes.

- *Dropout*: É uma técnica de regularização que escolhe aleatoriamente alguns neurónios durante o treino da rede e os desativa temporariamente, evitando *Overfitting*.

3.3 Arquitetura do Modelo

Para realizar as previsões horárias de temperatura do ar foi desenvolvido um modelo de aprendizado profundo utilizando RNA recorrentes do tipo LSTM na linguagem de programação Python, com base no trabalho de Chollet (2021), com foco em previsões *multi-step* diretas, para evitar o acúmulo de erros (Taieb; Sepp et al., 2012), principalmente para previsões de longo prazo. Isso significa que o modelo prevê todo o horizonte futuro de uma só vez. O editor de código utilizado para compilar o algoritmo foi o Visual Studio Code (VS-Code) disponibilizado na ferramenta de software de programação Anaconda. O dispositivo onde o modelo foi desenvolvido e executado tem as seguintes características:

- Processador: 13th Gen Intel(R) Core (TM) i5-13420H 2.10 GHz.
- RAM: 8 GB
- Sistema operacional de 64 bits, processador baseado em x64.

Os hiperparâmetros do modelo LSTM foram inicialmente definidos com base no trabalho de (Chollet, 2021). Posteriormente, foram realizados ajustes empíricos, com o aumento de alguns valores, visando melhorar a capacidade de aprendizado do modelo com um custo computacional razoavelmente baixo. Para a leitura e processamento dos dados, o modelo utiliza as bibliotecas *numpy* e *pandas*, e o *framework Keras*. O código lê o arquivo CSV contendo os dados horários das quatro variáveis meteorológicas, remove os valores ausentes e normaliza os dados entre 0 e 1, processo necessário em modelos de redes neurais por utilizar funções de ativação sigmoide e por melhorar a convergência do otimizador.

Os dados foram divididos em 60% para treinamento do modelo, 20% para validação e 20% para teste. Essa divisão é temporal, e não aleatória, pois é necessário preservar a causalidade da série. Caso essa divisão fosse aleatória, poderia acontecer do modelo vê o futuro durante o processo de treinamento, causando um vazamento de informações do futuro para o passado. Além disso, a divisão temporal é a que simula o mundo real, já que são necessárias informações históricas disponíveis para realizar previsões futuras.

O modelo é treinado por 100 épocas com um *batch size* de 64 amostras. Sua Arquitetura é composta por duas camadas LSTM com 64 neurônios cada. Também inclui camadas *dropout* entre elas para evitar *overfitting*. Utiliza como função de perda no processo de validação o Erro Quadrático Médio (MSE), próprio para problemas de regressão, e o otimizador Adaptive Moment Estimation (Adam) (Kingma; Ba, 2015) para fazer o ajuste automático dos pesos.

O otimizador Adam é um método amplamente utilizado no treinamento de RNR. Ele combina as vantagens de métodos baseados em média móvel do gradiente e da variância do gradiente para adaptar dinamicamente a taxa de aprendizado de cada parâmetro (Kingma; Ba, 2015). Ele calcula estimativas do primeiro e do segundo momento do gradiente. Essas estimativas são definidas pelas equações 9 e 10:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (9)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (10)$$

onde g_t representa o gradiente no instante t , enquanto β_1 e β_2 são coeficientes de decaimento exponencial utilizados para controlar a média móvel dos momentos.

A atualização dos pesos do modelo é então realizada por meio da equação 11:

$$\theta_{t+1} = \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (11)$$

em que α representa a taxa de aprendizado, \hat{m}_t e \hat{v}_t correspondem às estimativas corrigidas de viés do primeiro e segundo momentos, e ϵ é uma pequena constante utilizada para garantir estabilidade numérica.

Esse mecanismo contribui para acelerar a convergência do processo de otimização e estabilizar o treinamento do modelo.

De acordo com literatura, a definição da janela de entrada (*look back*) deve considerar a presença de ciclos sazonais relevantes, de modo que o modelo seja capaz de aprender dependências temporais representativas do processo analisado (Hyndman; Athanasopoulos, 2021; Box et al., 2015). Diante disso, para realizar as previsões de menor prazo (até 7 dias) o modelo foi treinado utilizando um *look back* de 168 horas (7 dias), escolhido com base no menor custo computacional, sem prejudicar o desempenho do modelo ao mesmo tempo que permite representar adequadamente a dinâmica semanal da série. Já para as previsões de 30 dias o modelo é treinado com um *look back* de 720 horas (30 dias) para que aprenda padrões mensais. Na etapa de teste, os valores do *look back* são os mesmos.

O valor do *look back* tem influência significativa no tempo de treinamento do modelo. Com o *look back* de 168 horas o modelo levou em média 1 dia e meio para

finalizar todo o processo de treinamento. Enquanto, com o valor de *look back* de 720 horas ele levou em torno de 2 dias.

Com o modelo treinado, as previsões são realizadas utilizando os dados de teste. Após esse processo, os dados resultantes são desnormalizados retornando a escala original. É extraída uma previsão com a quantidade de horas determinada, tanto dos valores observados quanto dos previstos, plotados em um gráfico. Os valores obtidos das métricas de avaliação utilizadas aparecem no terminal. Além disso, para cada horizonte de previsão determinada, um arquivo no formato Excel é gerado contendo os valores numéricos reais e previstos.

3.4 Métricas de Avaliação

Uma das etapas mais importantes na realização de previsões é avaliar o desempenho do modelo e a qualidade dos resultados obtidos. Para avaliar a qualidade das previsões do modelo neural LSTM dessa pesquisa utilizam-se a métrica de avaliação Raiz do Erro Médio Quadrático, em inglês *Root Mean Squared Error* (RMSE) e o erro médio absoluto, em inglês *Mean Absolute Error* (MAE), pois são comumente usadas na avaliação do desempenho de modelos de previsão e são calculados na mesma unidade de medida da saída do modelo, sendo de fácil interpretação (Moriasi et al., 2015). Além dessas métricas, a média do viés de cada previsão também será calculada para qualificar a avaliação.

O **RMSE** é uma das métricas mais utilizadas em modelos de regressão estatística (Hyndman; Koehler, 2006). Dentre as métricas, é a que realiza uma avaliação mais criteriosa sobre o desempenho do modelo. Ela mostra em média o quão distante os valores previstos estão dos observados (verdade terrestre). O mais comum de se encontrar na literatura científica é afirmação de que seus valores não têm limite superior e quanto mais próximo do zero melhor o desempenho do modelo e a precisão das previsões (Binoti et al., 2015). Porém, de acordo com Moriasi et al. (2015) e Singh et al. (2005) um RMSE é considerado baixo quando seu valor é inferior à metade do desvio padrão dos valores observados. O RMSE é calculado usando a equação 12:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_i - f_i)^2} \quad (12)$$

onde d_i representa a verdade terrestre, f_i os valores previstos e n a quantidade de amostras.

O Desvio Padrão s é uma medida de dispersão dos dados relativamente à média (Martins, 2013). É obtida calculando a raiz quadrada da variância amostral, ilustrada pela expressão 13. A metade do desvio padrão define o limite aceitável da previsão em relação ao valor do RMSE (Moriasi et al., 2015).

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (13)$$

onde x_i é o i -ésimo valor observado, \bar{x} é a média aritmética dos dados, expressa pela equação 14, e n é o número total de observações da amostra.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (14)$$

O **MAE** é uma métrica de avaliação utilizada em problemas de previsão, que calcula a média dos valores absolutos dados pela diferença entre os valores previstos e a verdade terrestre. Assim como o RMSE, também não possui limite máximo, e quanto mais próximo de zero, melhor o desempenho do modelo. O MAE é dado pela equação 15:

$$MAE = \frac{1}{n} \sum_{i=1}^n |d_i - f_i| \quad (15)$$

onde n é o número de amostras, d_i são os valores da verdade terrestre e f_i os valores previstos.

O **viés médio** \bar{e} serve para medir a tendência sistemática de cada previsão e indica se estar superestimando ou subestimando os valores observados. É calculado pela equação 16 :

$$\bar{e} = \frac{1}{n} \sum_{i=1}^n (f_i - d_i) \quad (16)$$

Onde n é o número total de previsões, f_i é o valor previsto no instante i e d_i é o valor observado no instante i .

3.5 Procedimentos para Gerar as Previsões

O modelo LSTM foi treinado inicialmente com os dados do INMET para realizar as previsões horárias de temperatura do ar na escala Nowcasting (6 horas), bem como para horizontes de previsão de 1 a 7 dias e de 30 dias. Essas previsões foram geradas para as últimas horas dos dados de testes que correspondem ao mês de dezembro de 2024. Para cada horizonte de previsão, foram gerados gráficos individuais, calculados os erros médios e armazenados os valores numéricos previstos e observados em tabelas no Excel.

O mesmo processo de treinamento e os mesmos horizontes de previsão foram os utilizados para o conjunto de dados do ERA5. Porém, além do treinamento utilizando todo o conjunto de dados (2001-2024), outros processos foram realizados, sendo um deles separar em conjuntos de dados sazonais por estação, treinar o modelo com eles e comparar os resultados obtidos, além de comparar com as previsões geradas pelo

modelo ao ser treinado com o conjunto de dados completo.

Outro processo realizado com o conjunto de dados do ERA5 foi treinar o modelo utilizando apenas dados de uma variável: a temperatura do ar a 2 m da superfície, em vez das quatro anteriormente citadas: temperatura do ar a 2m da superfície, temperatura do ponto de orvalho a 2m, pressão à superfície e velocidade do vento a 10 m da superfície, e comparar os resultados obtidos

Após todas essas etapas, foi realizada, por fim, uma comparação das previsões obtidas com o modelo sendo treinado com o conjunto de dados do INMET e com os dados de reanálise do ERA5. Os passos para realizar essa comparação consistiu em: organizar os dados em uma tabela individual para cada período previsto e gerar um gráfico de linha. Essas tabelas possuem quatro colunas: a primeira contém a data e hora, a segunda os valores observados do INMET, a terceira os valores previstos, e a quarta e última contém os dados de reanálise do ERA5. O objetivo dessa comparação foi verificar a qualidade das previsões do modelo LSTM em relação aos dados de reanálise do ERA5, considerados, cientificamente, de alta qualidade.

4 RESULTADOS

Neste capítulo são apresentados os resultados obtidos das previsões horárias de temperatura do ar geradas pelo modelo LSTM ao ser treinado a partir de sete conjuntos de dados distintos, sendo dados observados do INMET e dados de reanálise do ERA5. Considerando os parâmetros adotados nesta pesquisa, explorou-se previsões em três escalas, nowcasting (6 horas), sinótica (entre 1 e 10 dias) e subsazonal (até 30 dias). Cada seção deste capítulo apresenta os resultados obtidos pelo modelo nos diferentes cenários de treinamento. Esses resultados são expressos em forma de tabelas e gráficos, permitindo a visualização do desempenho do modelo e da qualidade das previsões geradas ao longo do tempo.

4.1 Modelo Treinado com Dados do INMET

Os primeiros resultados obtidos foram gerados após o modelo LSTM ser treinado com os dados do INMET, gerando as previsões para os períodos de 6, 24, 48, 72, 96, 168 horas e 30 dias, referente ao mês de dezembro de 2024. O desempenho do modelo ao gerar essas previsões, pode ser analisado por meio dos valores das métricas de avaliação do desempenho do modelo apresentados na tabela 1.

Tabela 1 – Valores das Métricas de Avaliação para as previsões realizadas durante o mês de dezembro de 2024.

Período Prev.	RMSE (°C)	MAE (°C)	Viés Médio (°C)
6 horas	1,89	1,79	-1,79
24 horas	0,97	0,79	-0,51
48 horas	0,98	0,85	-0,50
72 horas	0,98	0,75	-0,19
96 horas	1,58	1,22	-0,03
168 horas	2,00	1,62	0,45
30 dias	2,94	2,40	1,39

Observa-se na tabela 1 que, para previsões de curto prazo, especialmente entre 24 e 72 horas, os valores de RMSE e MAE permanecem relativamente baixos, indicando

bom desempenho do modelo nas previsões geradas nesse intervalo. O menor erro ocorre na previsão de 24 horas, com RMSE de $0,97^{\circ}\text{C}$ e MAE de $0,79^{\circ}\text{C}$, sugerindo que o modelo apresenta maior capacidade preditiva nesse horizonte temporal. Os erros estáveis para o horizonte de 48 e 72 horas indicam que o modelo consegue manter consistência na previsão da temperatura ao longo de alguns dias. Além disso, o viés médio nesses períodos é relativamente pequeno (entre $-0,50^{\circ}\text{C}$ e $-0,19^{\circ}\text{C}$), demonstrando baixa tendência sistemática de superestimação ou subestimação do modelo ao gerar as previsões nesses períodos.

Entretanto, à medida que o horizonte de previsão aumenta, observa-se um declínio gradual da precisão do modelo, apresentando erros maiores a medida que o horizonte de previsão aumenta. Esse aumento progressivo dos erros é esperado em modelos de previsão temporal, pois a incerteza associada às condições atmosféricas tende a crescer conforme o horizonte de previsão se torna mais longo (Lorenz, 1963; Bednár; Kantz, 2022), mesmo com modelos avançados de deep learning (Lam et al., 2023).

A análise dos valores do viés médio indica que nos horizontes de curto prazo (6 a 96 horas) o modelo subestima a temperatura, apresentando valores negativos. Já a partir de 168 horas (sétimo dia) os valores crescem, e o modelo passa a superestimar sistematicamente os valores de temperatura, indicando dificuldades do modelo em representar de forma adequada a tendência da temperatura prevista ao longo do tempo.

Além da tabela apresentada, a qualidade dessas previsões pode ser verificada de forma mais detalhada na Figura 8, que apresenta o limite aceitável do RMSE para cada previsão.

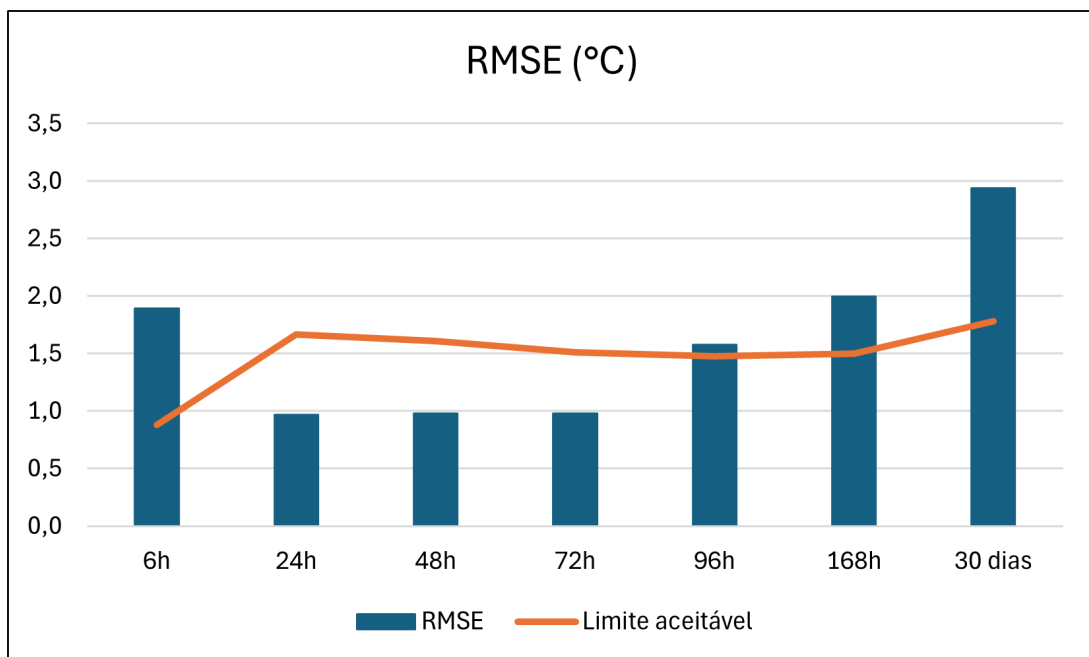


Figura 8 – Gráfico com os valores do RMSE e o seu limite aceitável para cada horizonte de previsão.

A análise dos valores da Tabela 1 e do gráfico da Figura 8, evidencia que o modelo apresentou melhor desempenho para as previsões horárias de temperatura do ar geradas para os horizontes de 24, 48 e 72 horas, nas quais o valor do RMSE permaneceu dentro do limite aceitável, assim como os valores do MAE e viés médio que obtiveram valores próximos de zero. Isso indica que o modelo LSTM treinado com dados observados do INMET é capaz de realizar previsões de menor prazo (até três dias) com boa precisão.

Em contrapartida não consegue manter o desempenho para previsões de curto prazo (até 6 horas) e para horizontes de previsões maiores que 72 horas, obtendo valores de erros médios elevados à medida que os horizontes de previsão aumentam, indicando baixa precisão nas previsões geradas nesses períodos. Dessa forma, verifica-se que a escala sinótica (1 a 3 dias) apresenta maior previsibilidade quando comparada à escala de nowcasting (até 6 horas) e à escala subsazonal (cerca de 30 dias). Uma possível explicação para o desempenho inferior na escala de nowcasting está relacionada às fases das previsões geradas pelo modelo, que são diurnas. Isto significa que o modelo apresenta melhor desempenho para ciclos diários completos em períodos curtos de até 3 dias.

A qualidade das previsões do modelo ao longo do tempo pode ser observada nas Figuras 9, 10, 11 correspondentes aos horizontes de previsão de 24 horas, 7 dias e 30 dias. O eixo das abscissas (eixo horizontal) corresponde à data e hora, expressa no formato mês–dia–hora, enquanto o eixo das ordenadas (eixo vertical) representa a temperatura do ar, em graus Celsius. Optou-se por apresentar apenas estes três resultados como forma de ilustração, sendo suficiente para verificar que as previsões do modelo, representadas pelo gráfico na cor vermelha, acompanha qualitativamente, de forma geral, a tendência dos dados observados, representada pelo gráfico na cor azul.

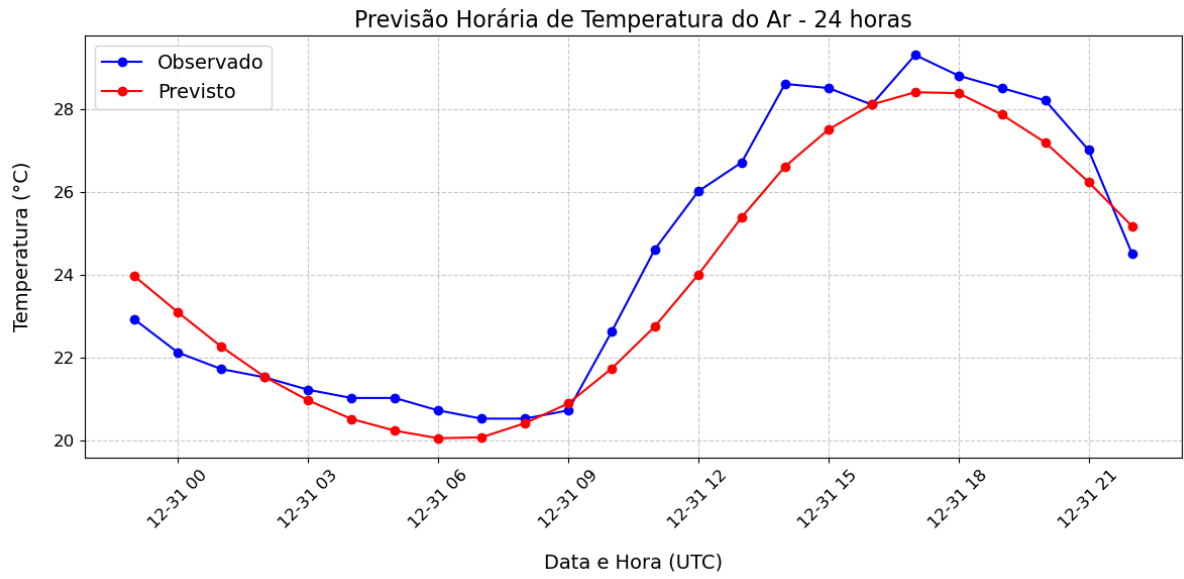


Figura 9 – Gráfico de previsão horária de temperatura do ar no período de 24 horas.

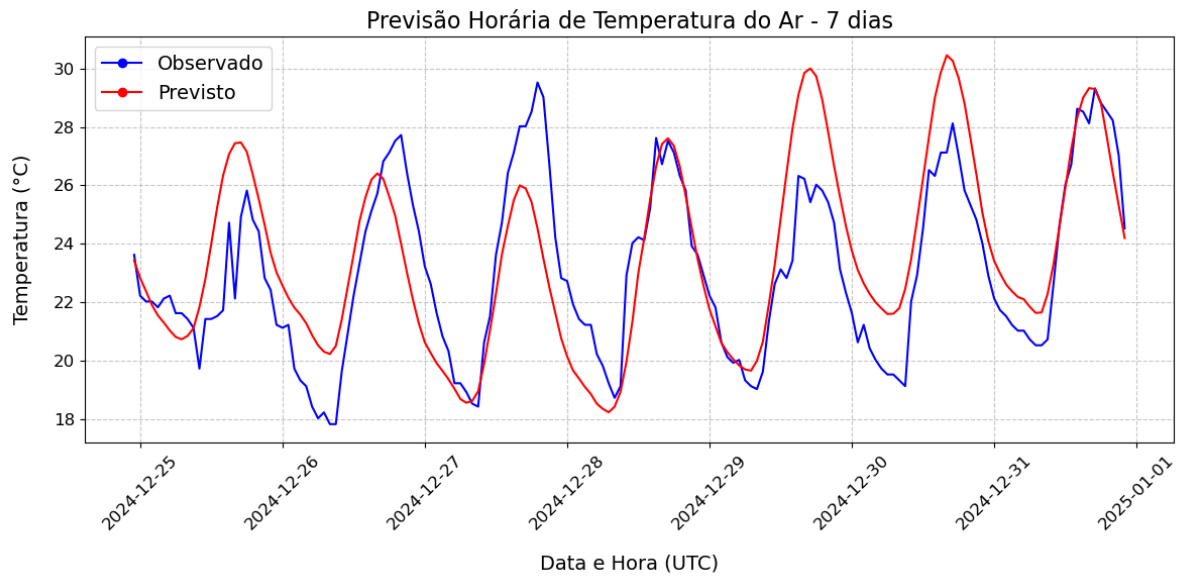


Figura 10 – Gráfico de previsão horária de temperatura do ar no período de 7 dias.

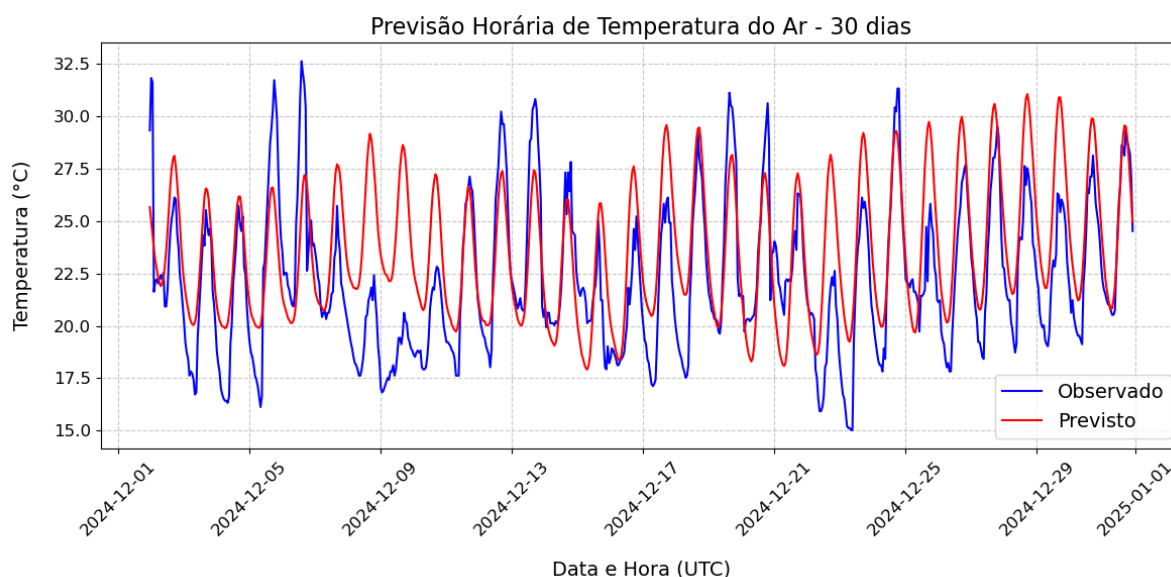


Figura 11 – Gráfico de previsão horária de temperatura do ar no período de 30 dias.

Pela análise das Figuras 9, 10, 11 é possível verificar que o modelo captou bem o padrão diário e conseguiu reproduzir o ciclo diurno (subida durante o dia e descida à noite) e a fase da onda dos valores previstos ficou bem próxima dos observados. Porém, nas mínimas o modelo subestima quedas mais extremas, o que indica que ele capta bem o padrão médio, mas não os ruídos de alta frequência, como queda súbita de temperatura por causa da chuva e rajada de vento frio. Isso acontece porque o modelo aprende o que é mais frequente e generaliza o padrão por meio da função de perda MSE, que faz o modelo aprender a média condicional da distribuição, levando a suavização das previsões (Bishop, 2006).

A dificuldade em prever extremos é um comportamento frequentemente observado em modelos de aprendizado profundo, incluindo o modelo com a LSTM, aplicados à previsão meteorológica, que tendem a reproduzir os padrões dominantes dos dados de treinamento, enquanto eventos extremos ou raros permanecem mais difíceis de prever com precisão (Olivetti; Messori, março, 2024).

4.2 Modelo Treinado com Dados do ERA5

Os resultados do modelo LSTM ao gerar as previsões horárias de temperatura do ar treinado com dados de reanálise do ERA5, são apresentados de forma semelhantes aos discutidos no tópico anterior (4.1), utilizando as mesmas variáveis e parâmetros. Porém, além dos resultados obtidos com o modelo sendo treinado com um conjunto de dados com várias variáveis, também foram analisados os resultados obtidos ao treiná-lo com apenas uma variável: a temperatura do ar, ou seja, usar temperatura para prever temperatura. Os resultados obtidos em cada um desses dois processos de treinamento são apresentados nos subtópicos a seguir.

4.2.1 Treinamento com Várias Variáveis Meteorológicas

O desempenho do modelo ao gerar as previsões horárias de temperatura do ar sendo treinado com o conjunto de dados de várias variáveis do ERA5, sendo elas: a temperatura do ar a 2 m da superfície, temperatura no ponto de orvalho, pressão à superfície e velocidade do vento à 10 m da superfície, e a qualidade dessas previsões podem ser analisadas pelos resultados apresentados na Tabela 2, contendo os valores dos erros médios, bem como pela figura 12, a qual contém os valores do RMSE e o limite teórico aceitável do valor do RMSE para cada período previsto.

Tabela 2 – Valores das Métricas de Avaliação para as previsões realizadas durante o mês de dezembro de 2024 com o modelo treinado com dados do ERA5.

Período Prev.	RMSE (°C)	MAE (°C)	Viés Médio (°C)
6 horas	0,57	0,51	-0,16
24 horas	0,80	0,65	-0,58
48 horas	0,84	0,68	-0,23
72 horas	0,95	0,79	0,40
7 dias	1,43	1,15	0,36
9 dias	1,55	1,27	0,36
10 dias	2,15	1,68	1,07
30 dias	3,65	3,06	2,61

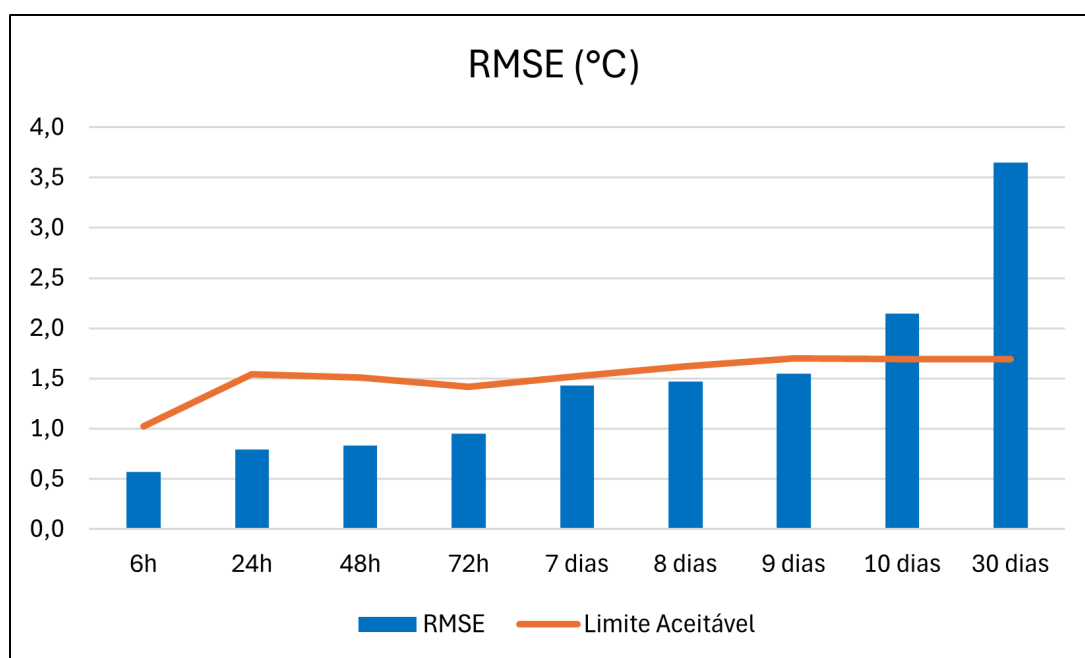


Figura 12 – Desempenho do modelo ao realizar previsões de temperatura do ar para diferentes horizontes de previsão ao ser treinado com dados do ERA5

Ao analisar a tabela 2 e o gráfico da Figura 12, verifica-se que os menores valores de erro ocorrem nas previsões de curto prazo. O modelo apresentou desempenho

excelente ao gerar as previsões para os horizontes de até 9 dias, superando significativamente o desempenho quando treinado com dados do INMET, que obteve boa precisão apenas para a escala sinótica (até 72 horas) (Figura 8). Além disso, obteve erros médios respectivamente menores para os horizontes de previsões maiores. Porém, assim como no caso anterior de treinamento, à medida que o horizonte de previsão aumenta, verifica-se um crescimento gradual nas métricas de erro (RMSE e MAE). No caso da previsão de 30 dias, os erros tornam-se substancialmente maiores, com RMSE de 3,65 °C e MAE de 3,06 °C, evidenciando a limitação do modelo LSTM, com os parâmetros usados nessa pesquisa, para gerar previsões de longo prazo.

A qualidade dessas previsões ao longo dos períodos de 24h, 7 dias e 30 dias também pode ser analisada nas Figuras 13, 14 e 15, onde se tem a previsão horária de temperatura ao longo do tempo, representado pelo gráfico verde, e os valores reais (dados de reanálise do ERA5) representado pelo gráfico laranja. O eixo das abscissas corresponde à data e hora, expressa no formato mês–dia–hora, enquanto o eixo das ordenadas representa a temperatura do ar, em graus Celsius.

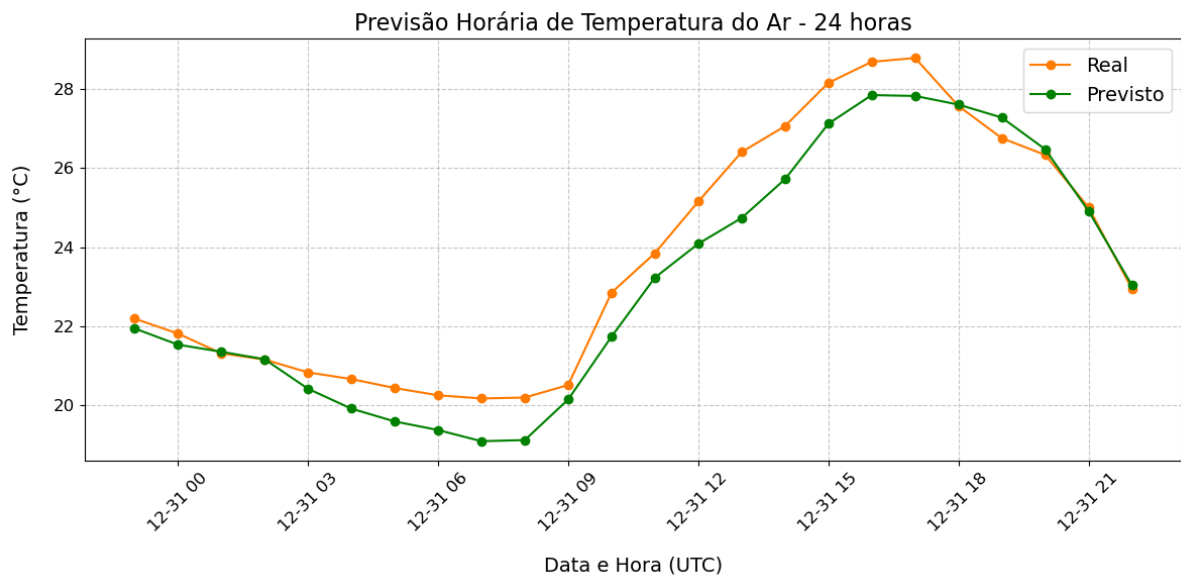


Figura 13 – Gráfico de previsão horária de temperatura do ar no período de 24 horas realizada ao modelo ser treinado com dados do ERA5

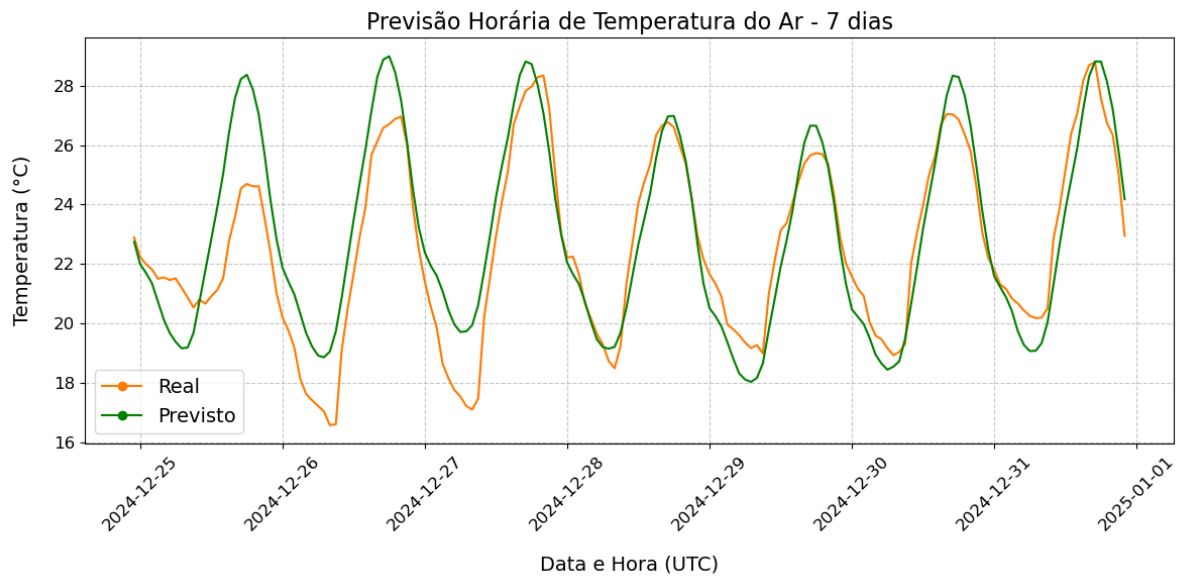


Figura 14 – Gráfico de previsão horária de temperatura do ar no período de 7 dias gerada pelo modelo ao ser treinado com dados do ERA5

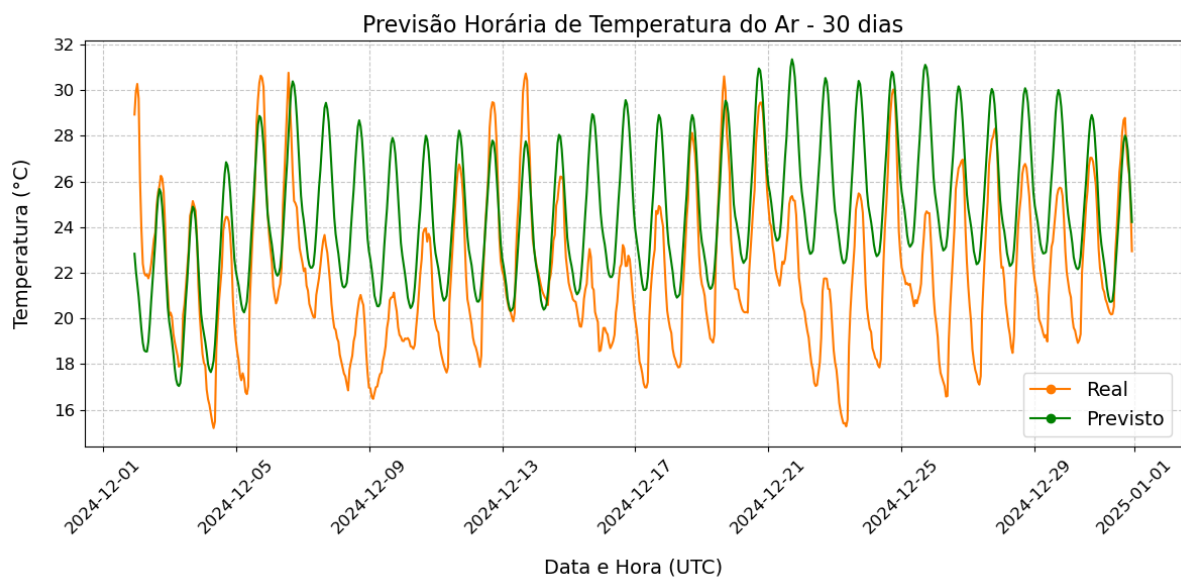


Figura 15 – Gráfico de previsão horária de temperatura do ar no período de 30 dias gerada pelo modelo ao ser treinado com dados do ERA5

Observa-se, pelas Figuras 13, 14 e 15, que as diferenças entre os valores reais e previstos tornam-se mais evidentes ao longo do tempo. Na primeira semana da previsão de 30 dias (Figura 15) os resultados mostram baixo erro de temperatura horária tanto em fase como em intensidade (inferiores a 2 graus Celsius). Isso deve-se a boa previsibilidade das variáveis utilizadas no conjunto de treinamento, tais como pressão, temperatura do ar, temperatura do ponto de orvalho e velocidade do vento, além da qualidade da série temporal utilizada (sem dados faltantes e sem valores anômalos). Os valores previstos (gráfico verde) apresentam variações mais suaves

em relação aos valores reais (gráfico laranja), concretizando o que já foi citado de que o modelo LSTM, com os parâmetros utilizados nessa pesquisa, apresenta dificuldade em prever os extremos, principalmente para períodos longos.

Outro ponto a ser notado é a diferença da amplitude térmica em vários momentos do gráfico. Essas divergências indicam que, assim como no caso anterior em que o modelo foi treinado com dados do INMET, o modelo superestima as temperaturas mínimas e subestima alguns picos de temperatura máxima, apesar de ter se saído melhor treinado com um conjunto de dados de reanálise do ERA5 em comparação aos resultados do subtópico anterior.

De maneira geral, os resultados demonstram que a rede LSTM treinada com dados de reanálise do ERA5 apresenta maior previsibilidade nos horizontes de curto e médio prazo (até 10 dias) em comparação com os resultados obtidos ao ser treinado com dados do INMET. Entretanto, nos dois casos, à medida que o horizonte de previsão aumenta, ocorre um crescimento consistente das métricas de erro (RMSE e MAE), evidenciando a diminuição da capacidade preditiva do modelo para escalas temporais mais longas (até 30 dias).

4.2.2 Comparação do Modelo treinado com uma Variável e com várias Variáveis Meteorológicas

Além de se avaliar o modelo neural LSTM treinado com as quatro variáveis meteorológicas: temperatura do ar a 2 m da superfície, temperatura no ponto de orvalho, pressão à superfície e velocidade do vento à 10 m da superfície da Reanálise ERA5, avaliou-se o mesmo treinado apenas com a temperatura do ar a 2 metros da superfície. A proposta inicial foi obter um modelo LSTM de previsão horária de temperatura do ar com baixa complexidade computacional e posteriormente, aumentar a complexidade, adicionando-se preditores. Essa comparação pode ser analisada no gráfico da figura 16, onde são apresentados os valores do RMSE referente a cada previsão.

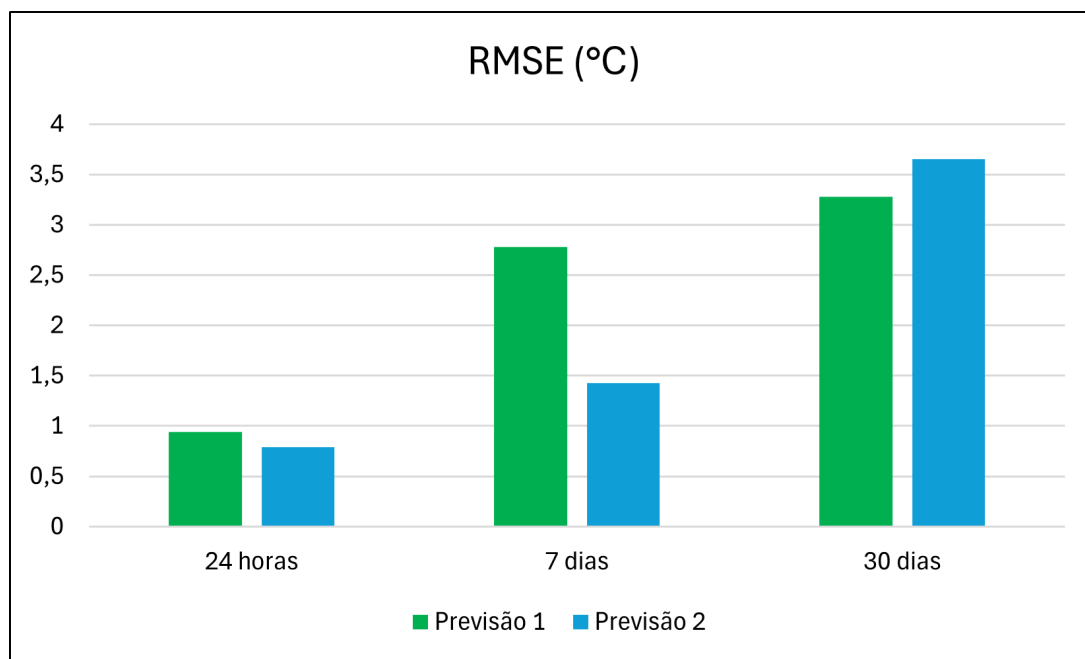


Figura 16 – Valores do RMSE das previsões geradas com o modelo LSTM treinado apenas com uma variável e ao ser treinado com várias variáveis

Os valores de RMSE correspondentes à "previsão 1", representados pelo gráfico na cor verde da figura 16, são referentes às previsões geradas pelo modelo ao ser treinado exclusivamente com dados horários de temperatura do ar, enquanto os valores da "previsão 2", representados pelo gráfico na cor azul, correspondem às previsões geradas com o modelo sendo treinado com as séries temporais referentes às quatro variáveis meteorológicas: temperatura do ar a 2 m da superfície, temperatura no ponto de orvalho, pressão à superfície e velocidade do vento à 10 m da superfície.

A análise da qualidade dessas previsões ao longo do tempo, comparando os resultados obtidos nos dois processos de treino, pode ser feita a partir da observação dos gráficos de linhas apresentadas nas Figuras 17, 18, 19. Ao realizar essa análise é possível verificar que as previsões geradas pelo modelo em ambos os treinamentos acompanham bem a tendência dos dados observados, principalmente para os horizontes de 24 horas e 7 dias. Porém, o modelo treinado apenas com os dados de temperatura não conseguiu acompanhar qualitativamente essa tendência para as previsões de longo prazo, isto é, para o período de 30 dias, evidenciado na Figura 19.

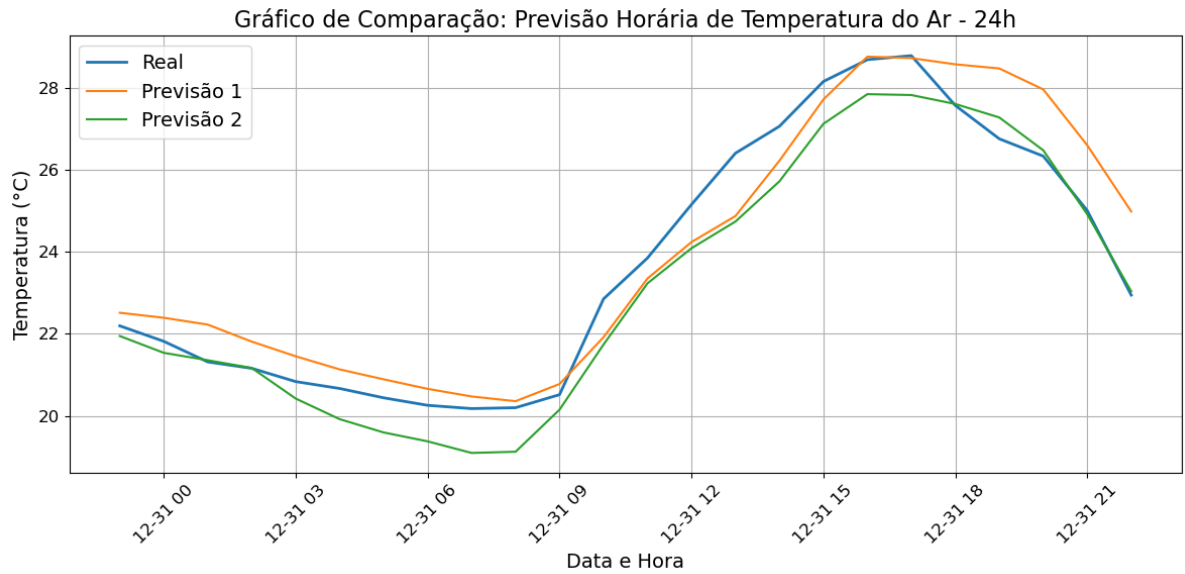


Figura 17 – Gráfico de comparação: previsões horárias de temperatura do ar no período de 24h geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas.

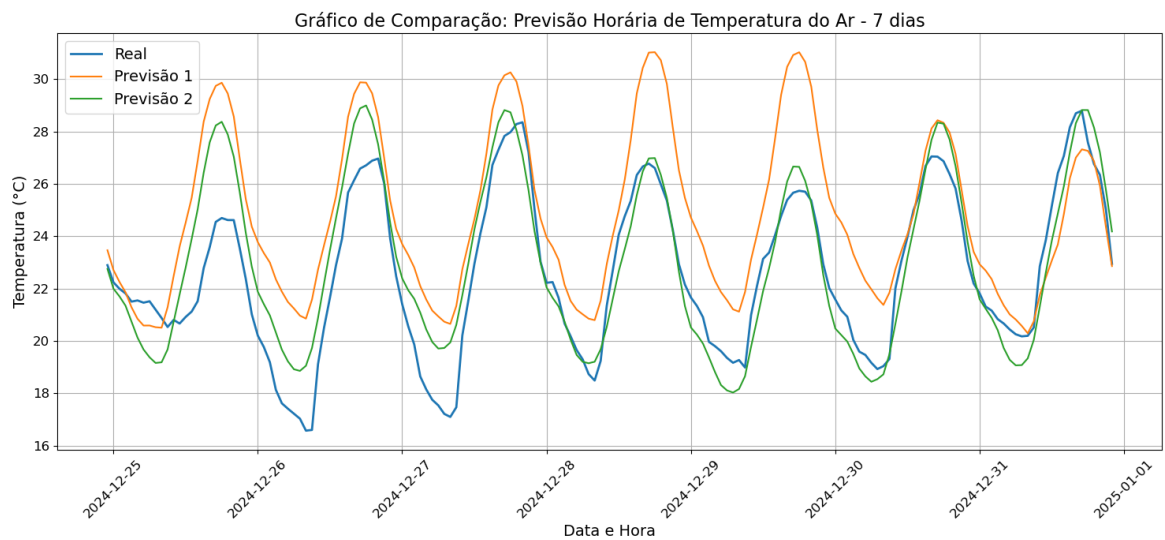


Figura 18 – Gráfico de comparação: previsões horárias de temperatura do ar no período de 7 dias geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas

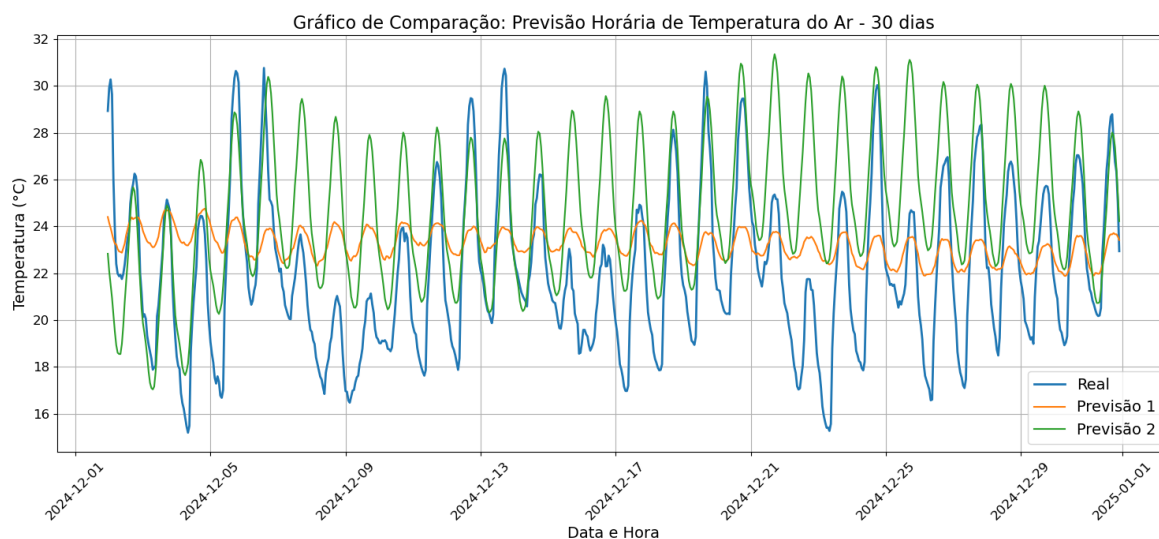


Figura 19 – Gráfico de comparação: previsões horárias de temperatura do ar no período de 30 dias geradas pelo modelo ao ser treinado com uma e com várias variáveis meteorológicas

Observa-se de forma geral nas Figuras 16, 17, 18 e 19, que as previsões geradas pelo modelo LSTM treinado com os 4 preditores gerou melhor resultado, representando melhor a tendência da temperatura horária do ar ao longo do tempo e obtendo valores do erro médio menores em comparação ao modelo treinado com apenas um preditor. A explicação está no fato de que a temperatura do ar é influenciada por múltiplos processos físicos, como transporte de calor, conteúdo de umidade e circulação atmosférica (Stull, 1988). Dessa forma, a inclusão de variáveis meteorológicas adicionais fornece informações complementares que permitem ao modelo capturar relações não lineares e interdependências entre os diferentes componentes do sistema atmosférico que influenciam diretamente na variação da temperatura do ar.

Ao se utilizar dados multivariados no treinamento do modelo LSTM, melhora a sua representação do sistema, nesse caso o comportamento da temperatura ao longo do tempo, devido a sua capacidade de explorar as correlações existentes entre diferentes parâmetros ambientais, resultando em melhor desempenho com erros médios menores quando comparado ao se utilizar apenas uma variável (Zhou et al., 2024; Yunantara; Daru, 2024). Isso evidencia a influência da qualidade e da quantidade do conjunto de treinamento na capacidade de generalização da rede LSTM.

4.3 Modelo Treinado com Dados Sazonais

Com o objetivo de verificar se o modelo teria um melhor desempenho ao ser treinado com dados padronizados, foram divididos quatro conjuntos de dados sazonais separados por estações do ano (verão, outono, inverno e primavera), no período de 2001 a 2024, para treinamento individual do modelo LSTM, utilizando os mesmos pa-

râmetros. O resultado correspondente ao desempenho do modelo após ser treinado separadamente por cada um desses conjuntos sazonais pode ser analisado no gráfico de barras da figura 20, contendo os valores do RMSE obtido das previsões horárias de temperatura do ar para os horizontes de 24 horas, 7 dias e 30 dias, além dos valores de RMSE obtidos nas previsões com o conjunto de dados completo, citados anteriormente, para fins de comparação.

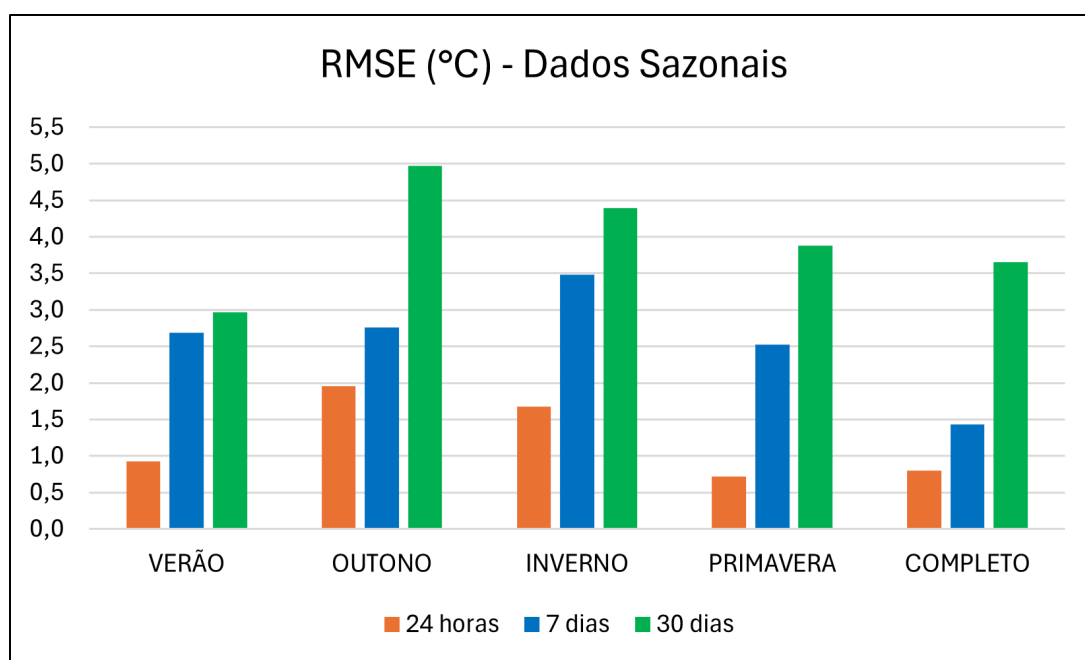


Figura 20 – Gráfico com os valores do RMSE das previsões referentes aos horizontes de previsão de 24h, 7 e 30 dias obtidas após o modelo ser treinado com os quatro conjuntos de dados sazonais

De acordo com os valores de RMSE apresentados na Figura 20, verifica-se que o modelo alcançou melhor desempenho ao ser treinado com o conjunto de dados completo (com as 4 estações do ano), em comparação com o treino realizado com os conjuntos de dados sazonais, pois apresentou valores de RMSE menores, indicando previsões com maior precisão. Isso indica que a utilização de um conjunto de dados mais abrangente permite que a rede capture uma maior diversidade de padrões atmosféricos, aumentando sua capacidade de generalização.

Esse comportamento pode ser explicado pelo fato de que conjuntos de dados mais diversos permitem que a rede LSTM capture uma maior variedade de padrões atmosféricos e relações não lineares presentes na dinâmica da temperatura do ar. Consequentemente, o modelo torna-se mais robusto e apresenta maior capacidade de generalização quando aplicado a diferentes condições meteorológicas, cenário proporcionado ao se juntar os dados das quatro estações. De acordo com Goodfellow et al. (2016), modelos de aprendizado profundo tendem a apresentar melhor desempenho quando treinados com conjuntos de dados maiores e mais diversos, pois isso permite uma representação mais completa da variabilidade do sistema estudado. Dessa

forma, conjunto completo ter tido os melhores resultados sugere que a inclusão de dados de diferentes estações do ano contribui para que a rede LSTM aprenda tanto padrões sazonais quanto as transições entre eles, possibilitando previsões com fases mais precisas ao longo do tempo.

Ao analisar individualmente os valores de RMSE de cada um dos quatro cenários sazonais, verifica-se que as previsões de menor prazo (1 e 7 dias) na primavera resultou em valores menores, indicando melhor desempenho do modelo e maior precisão das previsões nessa estação para esses períodos. Já para as previsões de longo prazo (30 dias), o menor RMSE obtido foi no verão, que, de forma geral, foi a estação em que o modelo apresentou o melhor desempenho em comparação com as outras. Isso pode estar associada à variabilidade das condições atmosféricas em cada estação e à quantidade de padrões presentes nos dados de treinamento. Durante o verão a temperatura do ar tende a apresentar maior regularidade no ciclo diário, o que significa fases mais suaves e consistentes, características próprias das previsões realizadas pelo modelo LSTM dessa pesquisa até o momento.

4.4 Comparação do modelo LSTM com o ERA5

Além de analisar o desempenho do modelo com as métricas de avaliação citadas, buscou-se realizar uma comparação gráfica com os dados de reanálise do ERA5 com o objetivo de verificar se as previsões horárias de temperatura do ar realizadas pelo modelo LSTM, treinado com dados observados do INMET, se aproximam dos valores dos dados de reanálise do ERA5. O resultado dessa comparação pode ser observado nas figuras 21, 22, 23, onde mostram-se os gráficos referente aos valores nos períodos de 24 horas, 7 dias e 30 dias. A linha verde representa os valores previstos, a linha azul os valores observados do INMET e a linha laranja os valores dos dados de reanálise do ERA5 referente a temperatura do ar a 2m da superfície.

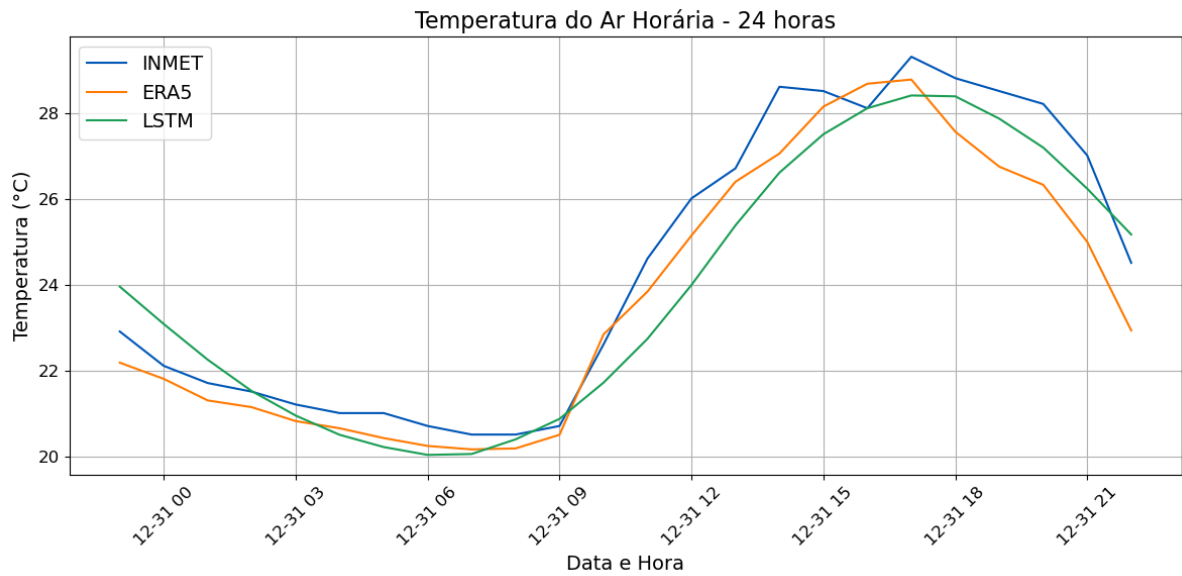


Figura 21 – Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 24 horas

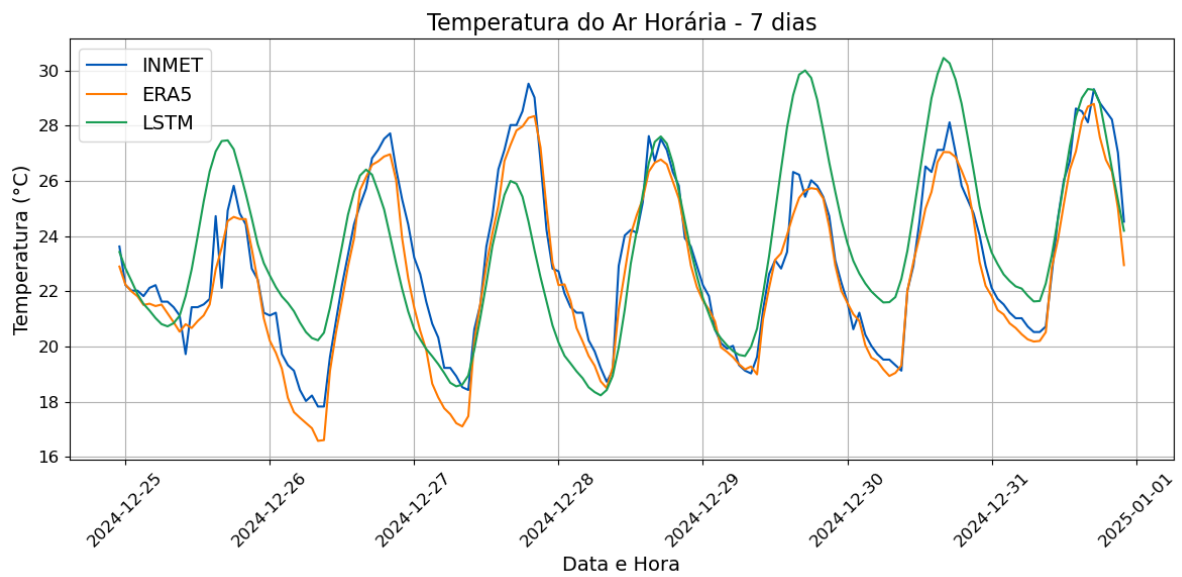


Figura 22 – Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 7 dias

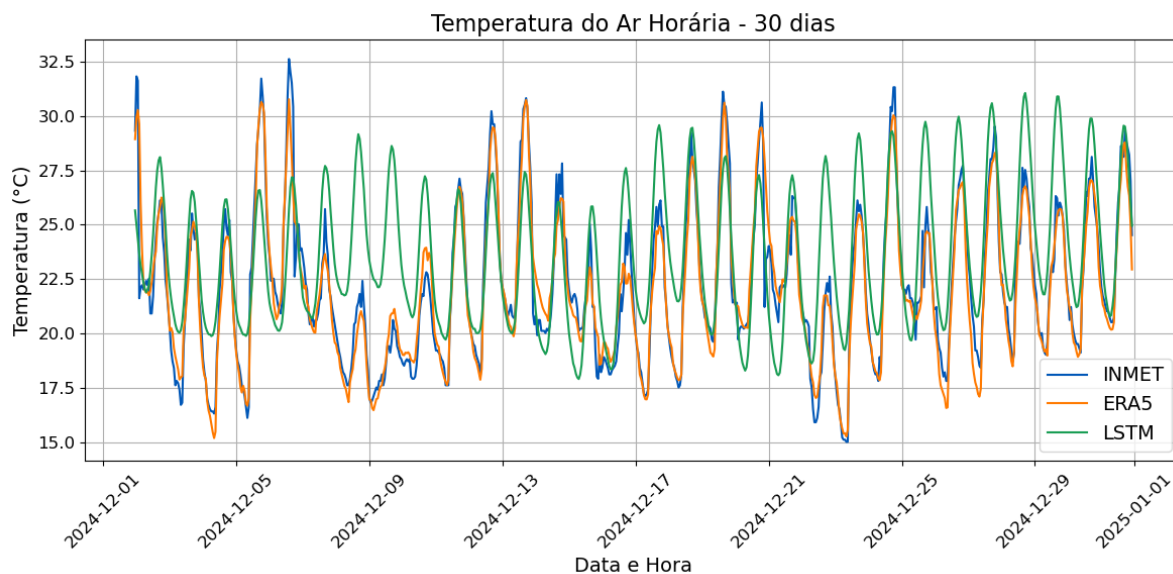


Figura 23 – Gráfico de comparação das previsões horárias de temperatura do ar do modelo LSTM com os dados de reanálise do ERA5 no período de 30 dias

Observa-se nas figuras 21, 22, 23 que tanto o ERA5 quanto o modelo LSTM reproduzem adequadamente a variabilidade temporal da temperatura, especialmente o ciclo diurno. Entretanto, os erros aumentam significativamente nos horários de máxima, especialmente a partir do sétimo dia de previsão. Isso evidencia que a previsão LSTM tende a superestimar mínimas e subestimar máximas, enquanto o ERA5 representa bem esses extremos e apresenta melhor representação dos dados observados em todos os períodos. Entretanto, o ERA5, por ser reanálise e previsões a partir de reanálise, é uma ferramenta indisponível para a previsão operacional.

5 CONCLUSÕES

Os resultados obtidos nesta pesquisa foram satisfatórios, possibilitando um conhecimento aprofundado do potencial do modelo LSTM na previsão horária de temperatura do ar em diferentes escalas temporais (6h até 30 dias). O modelo LSTM, assim como modelos físicos/dinâmicos, apresentou melhor resultado na escala sinótica (entre 1 e 10 dias), com decaimento de previsibilidade à medida que aumenta o horizonte de previsão, sendo que o RMSE foi inferior a 1 grau Celsius nos primeiros 3 dias, com pequeno viés de aquecimento. Isto explica-se por se utilizar um conjunto de dados de treinamento de 24 anos, com boa generalização e variáveis adequadas.

Na escala de Nowcasting as redes neurais são uma alternativa bastante competitiva em relação aos modelos de equações primitivas, sendo que a LSTM treinada com reanálise, gerou previsões muito boas, com RMSE inferior a 0,5 °C, onde o aceitável é 1,0. É importante destacar que erros de temperatura, numa região de grande variabilidade como latitudes médias, são aceitavelmente maiores do que regiões tropicais, onde a temperatura tem menor variabilidade ao longo do dia e das estações do ano.

Outra conclusão importante, decorrente deste trabalho, foi obtido ao se treinar a LSTM com quatro preditores. Optou-se por variáveis que tem melhor *skill* e que sejam disponibilizadas no INMET e ERA5, para efeitos de comparação. A Rede Neural treinada com maior número de preditandos (*labels*) exige maior tempo de treinamento e maior capacidade de processamento computacional, o que não foi uma dificuldade determinante para previsão pontual. No entanto, o uso de quatro preditores gerou previsões mais precisas com erros médios menores em relação ao treinamento com apenas 1 preditor.

Ao se utilizar quatro redes, uma para cada estação do ano (primavera, verão, outono e inverno), obteve-se resultados inferiores ao se treinar uma única rede para as previsões nas 4 estações do ano. Isto significa que dados não similares implicam num modelo com maior capacidade de generalização.

Um das limitações da pesquisa foi ter utilizado poucos dados de entrada, poucas camadas e poucos neurônios. O MSE suaviza a fase da previsão ao longo do tempo e dificulta a previsão dos extremos. O modelo LSTM é baseado puramente em

dados e não leva em consideração as leis físicas que governam a atmosfera (como conservação de energia, dinâmica atmosférica ou processos radiativos). Assim, ele pode reproduzir padrões históricos, mas tem dificuldade em prever mudanças bruscas associadas a frentes frias, advecção de massas de ar ou eventos extremos.

O desempenho do modelo LSTM depende da escolha adequada dos parâmetros, como número de camadas, número de neurônios, quantidade de épocas, taxa de aprendizado e tamanho da janela temporal (look back). Escolhas inadequadas podem levar a overfitting ou underfitting, situação em que a rede não consegue aprender adequadamente os padrões presentes nos dados, reduzindo, assim, a capacidade de generalização do modelo.

Outro desafio é que a LSTM funciona como uma “caixa-preta”, dificultando a interpretação física de como cada variável meteorológica influencia a previsão da temperatura. Isso ocorre porque as redes neurais profundas realizam milhares ou milhões de operações matemáticas internas (multiplicações de pesos, funções de ativação e estados de memória). Essas operações transformam os dados ao longo de várias camadas e passos temporais, tornando difícil relacionar diretamente uma causa física a um resultado específico da previsão.

Embora o modelo consiga prever a temperatura de curto e médio prazo com boa precisão, não é simples interpretar como cada variável meteorológica contribui para o resultado. Técnicas de interoperabilidade, análise de sensibilidade e modelos híbridos são estratégias que ajudam a reduzir essa limitação. O recomendado é combinar o modelo LSTM com outros modelos capazes de incorporar informações físicas da atmosfera. Isso pode possibilitar a integração do conhecimento físico de outras ferramentas com a capacidade da LSTM de aprender padrões temporais complexos. Ao realizar essa combinação, horizontes de previsões mais longos podem ser explorados, incluindo escalas mensais, semestrais e até anuais, ampliando a capacidade preditiva do modelo.

Em pesquisas futuras, pretende-se explorar outras redes neurais e o impacto de cada preditor no resultado da previsão, bem como aplicar o modelo neural a outros preditandos, como umidade, radiação, precipitação e até sistemas meteorológicos, como Frentes Frias, Sistemas Convectivos de Mesoescala, Linhas de Instabilidade, entre outros. Pretende-se ainda, prever campos de variáveis ao invés de pontos, pois assim constrói-se um modelo neural completo.

Por fim, entende-se que os objetivos propostos nesta dissertação foram alcançadas e vislumbram-se novas possibilidades de pesquisa a partir deste trabalho.

REFERÊNCIAS

ARTAXO, P. S. et al. Inteligência artificial e mudanças climáticas. **Revista USP**, São Paulo, n.141, p.29–40, 2024.

AYOADE, J. O. **Introdução à Climatologia para os Trópicos**. 4.ed. Rio de Janeiro: Bertrand Brasil, 1996.

BEDNÁR, H.; KANTZ, H. Prediction error growth in a more realistic atmospheric toy model with three spatiotemporal scales. **Geoscientific Model Development**, Göttingen, v.15, p.4147–4165, 2022.

BENGIO, Y.; SIMARD, P.; FRASCONI, P. Learning long-term dependencies with gradient descent is difficult. **IEEE transactions on neural networks**, New York, v.5, n.2, p.157–166, 1994.

BI, K. S. et al. Accurate medium-range global weather forecasting with 3D neural networks. **Nature**, London, United Kingdom, v.619, n.7970, p.533–538, 2023.

BINOTI, M. L. M. d. S. S. et al. Prognose em nível de povoamento de clones de eucalipto empregando redes neurais artificiais. **Cerne**, Lavras, v.21, n.1, p.97–105, 2015.

BISHOP, C. M. *Pattern Recognition and Machine Learning*. , New York, 2006.

BOX, G. E. P. S. et al. **Time Series Analysis: Forecasting and Control**. 5.ed. Hoboken, NJ, USA: Wiley, 2015.

BROWNLEE, J. **Introduction to Time Series Forecasting with Python**. Melbourne (Australia): Machine Learning Mastery, 2020.

CASTRO, E. S. et al. Previsão de Temperatura da Superfície do Mar (TSM) no Atlântico Tropical com uso de Redes Neurais. **Revista Brasileira de Meteorologia**, São Paulo, v.35, n.3, p.505–515, jul. 2020.

CHEN, K. S. et al. Fengwu: Pushing the skillful global medium-range weather forecast beyond 10 days lead. **arXiv preprint arXiv:2304.02948**, Ithaca, NY, 2023.

CHEN, L. S. et al. FuXi: a cascade machine learning forecasting system for 15-day global weather forecast. **npj climate and atmospheric science**, London, v.6, n.1, p.190, 2023.

CHOLLET, F. **Deep Learning with Python**. New York: Simon and Schuster, 2021.

DHITAL, S. et al. Forecasting weather using deep learning from meteorological stations data: A study of different meteorological stations in Kaski District, Nepal. **Eurasian Journal of Science and Engineering**, Sarajevo (Bosnia and Herzegovina), v.10, n.2, p.16–33, 2024.

DÍAZ-RAMÍRE et al. Comparative analysis of machine learning techniques for forecasting weather: a case study. **Interciencia**, Caracas (Venezuela), v.49, n.5, p.305–313, 2024.

European Centre for Medium-Range Weather Forecasts. **ERA5 Reanalysis – Single Levels**. Acesso em: out. 2025, Copernicus Climate Change Service (C3S). Disponível em: <https://cds.climate.copernicus.eu/datasets/reanalysis-era5-single-levels>.

FRNDA, J. S. et al. ECMWF short-term prediction accuracy improvement by deep learning. **Scientific Reports**, London, v.12, n.1, p.7898, 2022.

FURNARI, L. S. et al. **AI algorithms to enhance weather forecasts over a topographically complex Mediterranean region**. Göttingen (Germany): Copernicus Meetings, 2025.

GOODFELLOW, I. S. et al. **Deep Learning**. Cambridge: MIT Press, 2016.

GOOGLE. **Instituto Nacional de Meteorologia**. Google Maps. Acesso em: 08 ago. 2025, <https://www.google.com/maps/place/Instituto+Nacional+de+Meteorologia/@-30.053045,-51.1750986,17z>.

GUO, S.; YANG, Y.; LIU, P. Assimilation of the deep learning-corrected global forecast system fields into the regional model for improving medium-range persistent precipitation forecasts. **Atmospheric Research**, Amsterdam, v.303, p.107318, 2024.

HAQUE, E.; TABASSUM, S.; HOSSAIN, E. A comparative analysis of deep neural networks for hourly temperature forecasting. **IEEE Access**, Piscataway, NJ, v.9, p.160646–160660, 2021.

HAYKIN, S. **Neural Networks and Learning Machines**. 3.ed. Upper Saddle River, NJ: Pearson, 2009.

HERSBACH, H. S. et al. The ERA5 global reanalysis. **Quarterly journal of the royal meteorological society**, London, v.146, n.730, p.1999–2049, 2020.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, Cambridge, v.9, n.8, p.1735–1780, 1997.

HOU, J. et al. Prediction of hourly air temperature based on CNN–LSTM. **Geomatics, Natural Hazards and Risk**, Londres, v.13, n.1, p.1962–1986, 2022.

HYNDMAN, R. J.; ATHANASOPOULOS, G. **Forecasting: Principles and Practice**. 3.ed. Melbourne (Australia): OTexts, 2021.

HYNDMAN, R. J.; KOEHLER, A. B. Another look at measures of forecast accuracy. **International journal of forecasting**, Amsterdam, v.22, n.4, p.679–688, 2006.

Instituto Nacional de Meteorologia. **Dados Históricos**. Acesso em: set. 2024, <https://portal.inmet.gov.br/dadoshistoricos>.

KAUR, A.; SHARMA, J.; AGRAWAL, S. Artificial neural networks in forecasting maximum and minimum relative humidity. **International Journal of Computer Science and Network Security**, Seoul (South Korea), v.11, n.5, p.197–199, 2011.

KEISLER, R. Forecasting global weather with graph neural networks. **arXiv preprint arXiv:2202.07575**, Ithaca, NY, 2022.

KINGMA, D. P.; BA, J. Adam: A Method for Stochastic Optimization. **International Conference on Learning Representations (ICLR)**, San Diego, 2015.

LAM, R. S. et al. Learning skillful medium-range global weather forecasting. **Science**, Washington, (USA), v.382, n.6677, p.1416–1421, 2023.

LORENZ, E. N. Deterministic Nonperiodic Flow. **Journal of the Atmospheric Sciences**, Boston, v.20, n.2, p.130–141, 1963.

MARTINS, M. E. G. Desvio padrão amostral. **Revista de ciência elementar**, Lisbon, v.1, n.1, p.20021, 2013.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, New York, v.5, n.4, p.115–133, 1943.

MORIASI, D. N. S. et al. Hydrologic and water quality models: Performance measures and evaluation criteria. **Transactions of the ASABE**, Piscataway (USA), v.58, n.6, p.1763–1785, 2015.

- NETO, H. L. S. et al. Aplicação de Redes Neurais Recorrentes em séries temporais de estações meteorológicas para imputação de dados: uma abordagem sobre micro-estações meteorológicas na região Oeste do Pará. , Porto Alegre, p.171–180, 2020.
- NGUYEN, T. S. et al. Climax: A foundation model for weather and climate. **arXiv preprint arXiv:2301.10343**, Ithaca, NY, 2023.
- OLAH, C. et al. Understanding lstm networks. , Mountain View (USA), 2015.
- OLIVETTI, L.; MESSORI, G. Advances and prospects of deep learning for medium-range extreme weather forecasting. **Geoscientific Model Development**, Munique (Germany), v.17, n.6, p.2347–2358, março, 2024.
- PATHAK, J. et al. Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators. **arXiv preprint arXiv:2202.11214**, Ithaca, NY, 2022.
- PERLEBERG, B. R. S. et al. Análise comparativa dos dados meteorológicos de Rio Brilhante MS: Estação INMET, Banco de dados BR-DWGD e Reanálise ERA5. **Ciência e Natura**, Santa Maria, v.47, n.esp. 3, p.e84142–e84142, 2025.
- PHAM, B. T. et al. Development of advanced artificial intelligence models for daily rainfall prediction. **Atmospheric Research**, Amsterdam, v.237, p.104845, 2020.
- RAMESH, K.; ANITHA, R. MARSpline model for lead seven-day maximum and minimum air temperature prediction in Chennai, India. **Journal of Earth System Science**, Bengaluru (India), v.123, n.4, p.665–672, 2014.
- ROWLAND, S. T. et al. Can ultra short-term changes in ambient temperature trigger myocardial infarction? , Reino Unido, v.143, p.105910, 2020.
- SALINAS, D. S. et al. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. **International journal of forecasting**, Amsterdam, v.36, n.3, p.1181–1191, 2020.
- SANTOS, A. F. dos; MATTOS, J. G. Z. de; KREBS, P. R. ESTUDO PRELIMINAR DA UTILIZAÇÃO DE REDES NEURAI NA PREVISÃO DE TEMPERATURA MÉDIA DIÁRIA PARA A CIDADE DE PELOTAS-RS. , Foz de Iguaçu, PR, Brasil, 2002.
- SCHIZAS, C. N. et al. Artificial neural networks in forecasting minimum temperature (weather). , Espoo (Finland), p.112–114, 1991.
- SCHULTZ, M. G. S. et al. Can deep learning beat numerical weather prediction? **Philosophical Transactions of the Royal Society A**, London, v.379, n.2194, p.20200097, abril, 2021.

SILVA, D. E. **Redes neurais artificiais e processamento de linguagem natural aplicados à previsão do minicontrato futuro do índice Ibovespa**. 2021. Dissertação (Mestrado em Ciência da Computação) — Centro Universitário da Fundação Educacional Inaciana (FEI), São Bernardo do Campo.

SINGH, J. S. et al. Hydrological modeling of the Iroquois river watershed using HSPF and SWAT 1. **JAWRA Journal of the American Water Resources Association**, Middleburg, v.41, n.2, p.343–360, 2005.

SØNDERBY, C. K. et al. Metnet: A neural weather model for precipitation forecasting. **arXiv preprint arXiv:2003.12140**, new York, 2020.

STULL, R. B. **An Introduction to Boundary Layer Meteorology**. Dordrecht (Netherlands): Springer, 1988.

TAIEB, S. B.; SEPP et al. A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition. **Expert systems with applications**, Amsterdam, v.39, n.8, p.7067–7083, 2012.

WILKS, D. S. **Statistical Methods in the Atmospheric Sciences**. 4.ed. San Diego: Academic Press, 2019.

YUNANTARA, C. N.; DARU, A. F. Optimalisasi Prediksi Parameter Lingkungan Menggunakan Model LSTM Multivariat dan Univariat. **Bulletin of Computer Science Research**, Medan (Indonesia), v.5, n.6, 2024.

YUNPENG, L.; DI, H.; JUNPENG, B.; YONG, Q. Multi-step ahead time series forecasting for different data patterns based on LSTM recurrent neural network. , Liuzhou (China), p.305–310, 2017.

ZHOU, K. S. et al. TE-LSTM: A Prediction Model for Temperature Based on Multivariate Time Series Data. **Remote Sensing**, Basel (Switzerland), v.16, n.19, p.3666, 2024.

Apêndices

APÊNDICE A – Algoritmo Utilizado Para Treinamento do modelo

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error, mean_absolute_error
from keras.models import Sequential
from keras.layers import LSTM, Dense, Dropout
import datetime

# -----
# Função para criar janelas diretas (previsão multi-step)
# -----
def create_direct_dataset(data, target_index, look_back, horizon):
    """
    Cria janelas de entrada (X) e saídas múltiplas (y) para previsão direta.
    - look_back: número de horas passadas usadas como entrada
    - horizon: número de horas futuras que o modelo tentará prever
    """
    X, y = [], []
    for i in range(len(data) - look_back - horizon):
        X.append(data[i:(i + look_back), :]) # entradas
        y.append(data[(i + look_back):(i + look_back + horizon), target_index]) #
    return np.array(X), np.array(y)

# -----
# Carregando e preparando os dados
# -----
data = pd.read_csv(
    r"caminho do arquivo,
    header=0,
    na_values="-9999",
    sep=",",

```

```

        encoding="latin1"
    )

data.columns = data.columns.str.strip()
variaveis = ["PRESSAO", "TEMPERATURA DO AR", "TEMPERATURA DO PONTO DE ORVALHO", "VE

dataset = data[variaveis].dropna().values
scaler = MinMaxScaler(feature_range=(0, 1))
dataset_scaled = scaler.fit_transform(dataset)

target_index = variaveis.index("TEMPERATURA DO AR")

# -----
# Divisão treino/validação/teste
# -----
train_size = int(len(dataset_scaled) * 0.6)
val_size = int(len(dataset_scaled) * 0.2)
test_size = len(dataset_scaled) - train_size - val_size

train = dataset_scaled[:train_size]
val = dataset_scaled[train_size:train_size + val_size]
test = dataset_scaled[train_size + val_size:]

# -----
# Criando datasets diretos
# -----
look_back = 720 #
horizon = 720 # previsão direta n horas à frente

X_train, y_train = create_direct_dataset(train, target_index, look_back, horizon)
X_val, y_val = create_direct_dataset(val, target_index, look_back, horizon)
X_test, y_test = create_direct_dataset(test, target_index, look_back, horizon)

print("Formato X_train:", X_train.shape)
print("Formato y_train:", y_train.shape)

# -----
# Modelo LSTM (saída multi-step)

```

```
# -----  
model = Sequential()  
model.add(LSTM(64, return_sequences=True, input_shape=(look_back, X_train.shape[2]))  
model.add(Dropout(0.2))  
model.add(LSTM(64))  
model.add(Dropout(0.2))  
model.add(Dense(horizon)) # saída dos valores (previsão direta)  
  
model.compile(loss='mean_squared_error', optimizer='adam')  
  
history = model.fit(  
    X_train, y_train,  
    epochs=100,  
    batch_size=64,  
    validation_data=(X_val, y_val),  
    verbose=2  
)  
  
model.save("modelo_lstm_previsao_direta_720h_720__100E_INMET.keras")
```

APÊNDICE B – Algoritmo Utilizado Para Gerar as Previsões

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from keras.models import load_model
import datetime
import os
from math import sqrt

# -----
# Função usada no treinamento
# -----
def create_direct_dataset(data, target_index, look_back, horizon):
    X, y = [], []
    for i in range(len(data) - look_back - horizon):
        X.append(data[i:(i + look_back), :])
        y.append(data[(i + look_back):(i + look_back + horizon), target_index])
    return np.array(X), np.array(y)

# -----
# Carregar dados
# -----
data = pd.read_csv(
    r"caminho do arquivo",
    header=0,
    na_values="-9999",
    sep=",",
    encoding="latin1"
)
data.columns = data.columns.str.strip()

#datas = pd.to_datetime(data["DATA_HORA"])

```

```

datas = pd.to_datetime(data["DATA_HORA"], format="%d.%m.%Y %H:%M:%S")

variaveis = ["PRESSAO", "TEMPERATURA DO AR",
             "TEMPERATURA DO PONTO DE ORVALHO", "VELOCIDADE DO VENTO"]

dataset = data[variaveis].dropna().values

# -----
# Normalização (igual ao treino)
# -----
scaler = MinMaxScaler(feature_range=(0, 1))
dataset_scaled = scaler.fit_transform(dataset)

target_index = variaveis.index("TEMPERATURA DO AR")

# -----
# Parâmetros
# -----
look_back = 720
horizon = 720

# -----
# Divisão 60% treino / 20% validação / 20% teste
# -----
train_size = int(len(dataset_scaled) * 0.6)
val_size    = int(len(dataset_scaled) * 0.2)

test_data   = dataset_scaled[train_size + val_size:]

# -----
# Criar as janelas do conjunto de TESTE
# -----
X_test, y_test = create_direct_dataset(test_data, target_index, look_back, horizon)

# -----
# A última janela do teste (mesma usada no Código 1)
# -----
X_input = X_test[-1].reshape(1, look_back, len(variaveis))

```

```

# -----
# Carregar modelo
# -----
model = load_model(
    r"caminho do arquivo")
#print(data["DATA_HORA"].head())
#print(data["DATA_HORA"].tail())

# -----
# Previsão
# -----
y_pred_test = model.predict(X_input)[0]

# --- determinar a quantidade de horas previstas---
y_pred_test = y_pred_test[:720]

y_real_test_norm = y_test[-1]

# -----
# Inverter normalização (forma correta)
# -----
temp_pred = np.zeros((horizon, len(variaveis)))
temp_prob = np.zeros((horizon, len(variaveis)))
temp_pred[:, target_index] = y_pred_test
temp_prob[:, target_index] = y_real_test_norm

inv_pred_test = scaler.inverse_transform(temp_pred)[:, target_index]
inv_real_test = scaler.inverse_transform(temp_prob)[:, target_index]

# -----
# Cálculo das métricas
# -----
errors = inv_pred_test - inv_real_test
RMSE = sqrt(np.mean(errors**2))
MAE = np.mean(np.abs(errors))
VIES = np.mean(errors)

```

```

print("\n===== MÉTRICAS (TESTE - JANELA FINAL) =====")
print(f"RMSE: {RMSE:.4f}")
print(f"MAE: {MAE:.4f}")
print(f"VIÉS: {VIES:.4f}")

# -----
# Datas reais correspondentes à última janela (ALINHAMENTO CORRETO)
# -----
datas_test = datas[train_size + val_size:].reset_index(drop=True)

# índice de início da última janela Y criada por create_direct_dataset()
# (ver derivação: a última i em range(len(data)-look_back-horizon) é i = len(data)-
inicio_ultima_janela = len(test_data) - horizon - 1

# pegar exatamente as datas correspondentes a essa janela
datas_horizonte = datas_test.iloc[inicio_ultima_janela : inicio_ultima_janela + hor

# -----
# Criar DataFrame com resultados
# -----
df_result = pd.DataFrame({
    "Data_Hora": datas_horizonte,
    "Observado": inv_real_test,
    "Previsto": inv_pred_test
})

# -----
# Salvar em Excel
# -----
caminho_saida = os.path.join(
    r"caminho do arquivo",
    f"previsao_direta_{horizon}h.xlsx"
)

df_result.to_excel(caminho_saida, index=False)
print(f"\n Resultados salvos em:\n{caminho_saida}")

```

```
# -----  
# Gráfico (estilo do Código 2)  
# -----  
plt.rcParams.update({  
    'font.size': 14,  
    'axes.titlesize': 16,  
    'axes.labelsize': 14,  
    'xtick.labelsize': 12,  
    'ytick.labelsize': 12,  
    'legend.fontsize': 14,  
})  
  
plt.figure(figsize=(12, 6))  
  
plt.plot(datas_horizonte, inv_real_test, 'o-', label='Real', color="#FF7A00")  
plt.plot(datas_horizonte, inv_pred_test, 'o-', label='Previsto', color='green')  
  
plt.title(f'Previsão Horária de Temperatura do Ar - {horizon} horas')  
plt.xlabel('Data e Hora (UTC)', labelpad=15)  
plt.ylabel('Temperatura (°C)', labelpad=15)  
  
plt.xticks(rotation=45)  
plt.legend()  
plt.grid(True, linestyle='--', alpha=0.7)  
plt.tight_layout()  
plt.show()
```