

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
DE SANTA CATARINA
CAMPUS SÃO JOSÉ

GUSTAVO PAULO

**APLICAÇÃO DE MODELOS FUNDAMENTAIS DE SÉRIES
TEMPORAIS PARA A PREVISÃO EM DIFERENTES DOMÍNIOS**

SÃO JOSÉ

2025

Gustavo Paulo

Aplicação de modelos fundamentais de séries temporais para a previsão
em diferentes domínios

Projeto de Trabalho de conclusão de curso
apresentado à Coordenadoria do Curso de
Engenharia de Telecomunicações do campus
São José do Instituto Federal de Santa Cata-
rina

Área de concentração: Telecomunicações

Orientador: Prof. Arliones Stevert Hoeller
Junior, Dr.

Coorientador: Prof. Mario de Noronha Neto,
Dr.

São José

2025

RESUMO

Diante da evolução dos Time Series Foundation Models (TSFM) e de sua capacidade de capturar dependências temporais complexas sem ajustes específicos, esses modelos vêm transformando pipelines de previsão em diversos domínios. A predição zero-shot dos TSFM elimina a etapa de coleta massiva de dados rotulados para treinamento, o que é especialmente vantajoso em aplicações onde os dados são escassos ou custosos de obter. Realiza-se levantamento do estado da arte e avaliação experimental de TSFM de código aberto em séries temporais provenientes de múltiplos domínios do Monash Time Series Forecasting Repository, quantificando seu desempenho por métricas como erro médio quadrático (MSE) e erro médio absoluto (MAE). Este trabalho visa aplicar e comparar esses modelos na previsão de séries temporais em diferentes setores, avaliando sua viabilidade e potencial estratégico.

Palavras-chave: TSFM. Séries temporais. Aprendizado profundo. Previsão de séries temporais.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diferença entre séries temporais univariáveis e multivariáveis	9
Figura 2 – Exemplo de diferença entre aprendizado supervisionado e não supervisionado	11
Figura 3 – Perceptron	13
Figura 4 – Comparação entre a arquitetura de um perceptron e de uma rede neural profunda	15
Figura 5 – Diferença entre modelos específicos e TSFM.	17

LISTA DE ABREVIATURAS E SIGLAS

DL Deep Learning.

LSTM Long Short-Term Memory.

MAE Mean Absolute Error.

ML Machine Learning.

MSE Mean Squared Error.

RNA Rede Neural Artificial.

RNN Rede Neural Recorrente.

TSFM Time Series Foundation Models.

SUMÁRIO

1	INTRODUÇÃO	6
1.1	OBJETIVO GERAL	6
1.2	OBJETIVOS ESPECÍFICOS	6
1.3	JUSTIFICATIVA	7
2	FUNDAMENTAÇÃO TEÓRICA	8
2.1	SÉRIES TEMPORAIS	8
2.1.1	Univariáveis e multivariáveis	8
2.1.2	Estacionariedade	9
2.1.3	Previsão de séries temporais	9
2.1.3.1	<i>Métricas de avaliação</i>	9
2.2	MACHINE LEARNING	10
2.2.1	Representation Learning	12
2.2.2	Redes Neurais	12
2.2.2.1	<i>Perceptron</i>	12
2.2.3	Deep Learning	14
2.2.3.1	<i>Evolução dos principais modelos de Deep Learning para previsão de séries temporais</i>	14
2.3	MODELOS FUNDAMENTAIS DE SÉRIES TEMPORAIS (TSFM)	17
2.4	MONASH TIME SERIES FORECASTING REPOSITORY	18
3	PROPOSTA	19
3.1	SELEÇÃO DOS CONJUNTOS DE DADOS, PRÉ-PROCESSAMENTO E ESCOLHA DOS MODELOS	19
3.2	PREVISÕES <i>ZERO-SHOT</i>	19
3.3	AJUSTE FINO	19
3.4	CONSTRUÇÃO DA PLATAFORMA WEB	19
3.5	CRONOGRAMA DE ATIVIDADES	20
	Referências	21

1 INTRODUÇÃO

Nos últimos anos, com a presença cada vez maior da tecnologia na vida das pessoas, consolidou-se um ambiente digital em que dados são gerados continuamente e em volumes cada vez maiores. Esse crescimento exponencial não apenas amplia os desafios de armazenamento e processamento, mas, sobretudo, impõe novas exigências analíticas para capturar padrões temporais em dados sequenciais, fundamentais em áreas como finanças, saúde, clima, etc (LIANG et al., 2024).

Diante dessa explosão de dados, os avanços do deep learning têm se mostrado revolucionários na modelagem de séries temporais. Arquiteturas como transformers emergiram como ferramentas potentes para lidar com dependências temporais complexas e variáveis exógenas, superando limitações dos métodos estatísticos tradicionais (CHEN et al., 2023).

Essa evolução culminou em uma nova fronteira de pesquisa: os Time Series Foundation Models (TSFM), inspirados nos modelos fundamentais que transformaram a visão computacional e o processamento de linguagem natural. Esses modelos representam uma mudança de paradigma ao permitir previsões zero-shot e few-shot a partir de pré-treinamento em larga escala, com notável generalização entre domínios e frequências temporais (LIANG et al., 2024).

A proposta desse projeto é aplicar e comparar diferentes Time Series Foundation Models (TSFM) de código aberto para previsão de séries temporais em múltiplos domínios, avaliando o desempenho dos principais modelos atuais por meio de métricas como Mean Squared Error (MSE) e Mean Absolute Error (MAE). Espera-se demonstrar que os TSFM podem ser modelos ágeis e eficientes em contextos onde os dados rotulados são escassos ou custosos de obter, obtendo resultados concretos e significativos.

1.1 OBJETIVO GERAL

O objetivo principal deste trabalho é aplicar diversos modelos de TSFM para previsão de séries temporais em múltiplos domínios.

1.2 OBJETIVOS ESPECÍFICOS

- Realizar uma pesquisa do estado da arte sobre TSFM.
- Coletar dados de séries temporais de diferentes domínios disponíveis no repositório Monash.
- Limpar e pré-processar a base de dados escolhida.

- Selecionar modelos de TSFM abertos para a realização de testes.
- Realizar o fine-tuning de pelo menos um modelo TSFM em um dos conjuntos de dados escolhidos.
- Realizar testes de diversos modelos na base de dados pré-processada.
- Desenvolver uma interface web simples que permita selecionar modelos TSFM e realizar/visualizar previsões.

1.3 JUSTIFICATIVA

A popularização recente dos modelos pré-treinados para séries temporais torna essencial investigar criticamente sua eficácia e capacidade real de generalização. Esses modelos possibilitam avaliar empiricamente o desempenho em diferentes cenários práticos sem necessidade de treinamento específico. Dessa forma, este trabalho se justifica por contribuir diretamente para a compreensão do potencial e limitações desses modelos em uma aplicação concreta.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 SÉRIES TEMPORAIS

Uma série temporal é uma coleção de observações ordenadas no tempo, comum em diversas áreas como economia, meteorologia, energia, etc. Ao contrário de conjuntos de dados independentes, os valores em uma série temporal apresentam dependência temporal, ou seja, o valor atual está frequentemente relacionado a valores passados. Em geral, séries temporais apresentam 4 principais padrões estruturais, que são tendência (mudanças de longo prazo), sazonalidade (flutuações regulares), ciclo (flutuações maiores que um ano) e ruído (variações imprevisíveis), o que influencia diretamente as abordagens de modelagem e previsão (REIS, 2023).

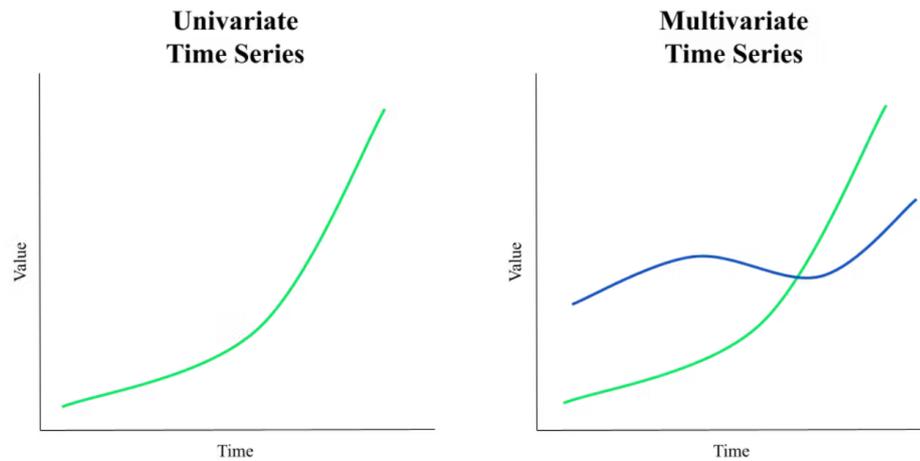
2.1.1 Univariáveis e multivariáveis

Uma série temporal pode ser formalmente representada como uma sequência de observações ordenadas no tempo, em que cada observação corresponde a um vetor de dimensão D . Quando $D = 1$, ou seja, cada ponto no tempo é composto por um único valor escalar, tem-se uma série temporal univariada. Um exemplo típico seria o monitoramento horário do preço de energia elétrica ao longo de vários dias (LIANG et al., 2024).

Por outro lado, quando $D > 1$, ou seja, cada observação é composta por múltiplos valores simultâneos, a série temporal é considerada multivariável (LIANG et al., 2024). Esse tipo de série permite representar cenários mais complexos, nos quais diversas variáveis evoluem juntas ao longo do tempo e podem apresentar relações de dependência entre si. Um exemplo seria o acompanhamento conjunto do preço da energia, da geração por usinas fotovoltaicas e da geração por usinas eólicas. Modelos multivariáveis permitem identificar relações entre variáveis que influenciam diretamente a qualidade das previsões. (MONTGOMERY; JENNINGS; KULAHCI, 2015).

Na Figura 1a é possível visualizar uma série temporal de apenas uma variável, já na Figura 1b a série temporal apresenta duas variáveis, sendo considerada multivariável por apresentar mais de um valor observado ao longo do tempo.

Figura 1 – Diferença entre séries temporais univariáveis e multivariáveis



(a) Série temporal univariável

(b) Série temporal multivariável

Fonte: TIME... ([s.d.])

2.1.2 Estacionariedade

A estacionariedade é uma propriedade estatística fundamental em muitas abordagens de modelagem de séries temporais. Uma série é dita estacionária quando suas propriedades, como média, variância e autocorrelação, permanecem constantes ao longo do tempo. (MONTGOMERY; JENNINGS; KULAHCI, 2015)

2.1.3 Previsão de séries temporais

A previsão de séries temporais é muito utilizada em diversas aplicações reais, como antecipar a demanda por energia, prever preços de mercado ou estimar o número de acessos a um site. No entanto, prever o futuro com base em dados passados envolve desafios como a presença de ruído, mudanças estruturais e limitações na disponibilidade de dados futuros. Para realizar essa previsão, metodologias baseadas em aprendizado de máquina (ML) têm ganhado cada vez mais relevância (DAS et al., 2024).

2.1.3.1 Métricas de avaliação

Ao comparar a previsão feita com os valores reais de uma série temporal, o objetivo é medir o quão longe a previsão ficou da realidade. Para isso, para cada passo temporal t , obtém-se o erro $e_t = y_t - \hat{y}_t$ (valor real menos valor previsto) então, resume-se todos esses erros em um único número que facilite a interpretação. Para isso, as duas medidas mais utilizadas são o Erro Quadrático Médio (MSE) e o Erro Absoluto Médio (MAE). O MSE eleva cada erro ao quadrado (e_t^2) antes de tirar a média, penalizando valores discrepantes de forma mais intensa. Quanto menor o MSE, melhor a predição. O MAE, por sua vez, usa o valor absoluto $|e_t|$ de cada erro e faz a média, expressando o desvio médio em unidades da

própria série e sendo menos sensível a valores extremos. Em ambos os casos, um resultado igual a zero indicaria uma previsão perfeita, enquanto valores maiores apontam uma maior diferença entre a previsão e a realidade (MONTGOMERY; JENNINGS; KULAHCI, 2015). Complementando essas métricas, será utilizada uma previsão de referência com o método Naïve, que consiste em repetir o último valor observado, para que seja possível comparar o desempenho dos modelos com um método de previsão simples e de fácil implementação.

2.2 MACHINE LEARNING

Aprendizado de máquina, do inglês Machine Learning (ML), é um ramo da inteligência artificial que se refere a sistemas capazes de aprender com base em padrões encontrados em dados, sem necessariamente serem programados para cada situação. Esses sistemas de ML são amplamente utilizados para realizar previsões ou tomar decisões com base em novas informações (MURPHY, 2012).

De forma geral, o ML é dividido em dois tipos principais:

- **Supervisionado:** Modelos supervisionados exigem que os dados de treinamento venham acompanhados de rótulos, ou seja, informações sobre a saída esperada para cada entrada. Durante o processo de treinamento, são mostrados ao modelo exemplos de dados junto com seus respectivos rótulos, permitindo que ele aprenda a relação entre entrada e saída (MURPHY, 2012).

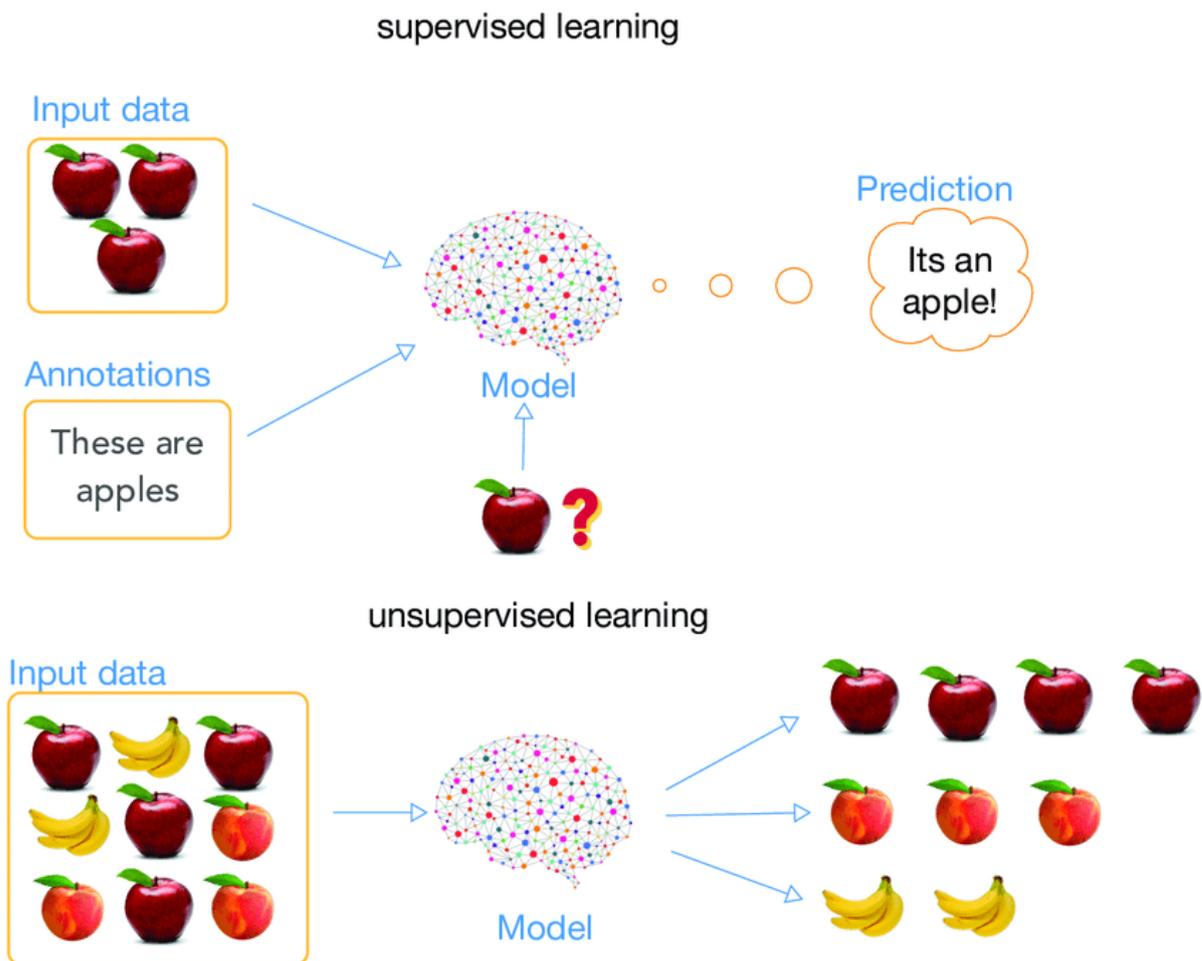
Por exemplo, imagine que o objetivo é classificar séries temporais em três categorias: A, B ou C. Para treinar o modelo, são fornecidas diversas séries já rotuladas com suas respectivas categorias, permitindo que ele aprenda a distinguir entre os diferentes tipos.

- **Não supervisionado:** Modelos não supervisionados, por outro lado, não utilizam rótulos nos dados. Em vez disso, eles buscam identificar estruturas ocultas ou padrões nos dados, como agrupamentos ou distribuições (MURPHY, 2012). Um caso de uso comum é a detecção de anomalias. Suponha que um conjunto de dados é composto por diversas séries temporais com comportamento estável e o objetivo é identificar quando ocorre uma anomalia. Dessa forma, um modelo não supervisionado pode aprender o padrão de estabilidade dessas séries durante o treinamento e, quando encontrar uma nova série que fuja desse padrão, reconhecê-la como anômala, mesmo sem nunca ter sido treinado com exemplos explícitos de anomalias.

A Figura 2 ilustra a diferença entre aprendizado supervisionado e não supervisionado. Na parte superior da imagem, o modelo recebe dados de entrada acompanhados de rótulos, que no exemplo ilustrado é a indicação dos objetos serem maçãs. Esse processo

caracteriza o aprendizado supervisionado, pois o modelo aprende a mapear entradas para saídas conhecidas. Por outro lado, a parte inferior da figura representa o aprendizado não supervisionado, no qual o modelo recebe apenas os dados de entrada, sem rótulos. Nesse caso, o objetivo do modelo é identificar padrões ou agrupamentos naturalmente presentes nos dados, como organizar diferentes frutas por similaridade visual, mesmo sem ter recebido rótulos de cada uma, ensinando exemplos de cada fruta previamente.

Figura 2 – Exemplo de diferença entre aprendizado supervisionado e não supervisionado



Fonte: Devopedia (2022)

Além dos tipos clássicos supervisionado e não supervisionado, existem variações que combinam ou estendem essas abordagens. Uma delas é o aprendizado auto-supervisionado, muito utilizado em modelos grandes como os modelos pré-treinados que serão abordados neste trabalho. Nessa estratégia, os próprios dados geram sinais de aprendizagem por meio de rótulos artificiais criados automaticamente ao mascarar partes de uma sequência, prever o próximo trecho ou comparar visões diferentes da mesma amostra, permitindo que o modelo aprenda representações úteis sem nenhuma anotação humana (BERGMANN, 2023).

2.2.1 Representation Learning

Representation Learning é um conceito essencial em ML e refere-se à capacidade dos modelos de aprender automaticamente representações ou características úteis diretamente a partir dos dados brutos (LECUN; BENGIO; HINTON, 2015). Isso é particularmente importante em problemas complexos, como séries temporais onde padrões temporais e variações podem ser difíceis de capturar de maneira tradicional.

2.2.2 Redes Neurais

As redes neurais artificiais (RNAs) são modelos computacionais inspirados no funcionamento paralelo e distribuído do cérebro humano. Elas têm como característica principal a capacidade de aprender padrões e relações complexas a partir de dados, generalizando seu aprendizado para novas situações. Isso as torna especialmente eficazes em tarefas como classificação, regressão e reconhecimento de padrões, que exigem modelagem não linear e adaptativa (HAYKIN, 1998).

De acordo com Goodfellow, Bengio e Courville (2016), uma RNA é composta por alguns elementos fundamentais que colaboram para o seu funcionamento:

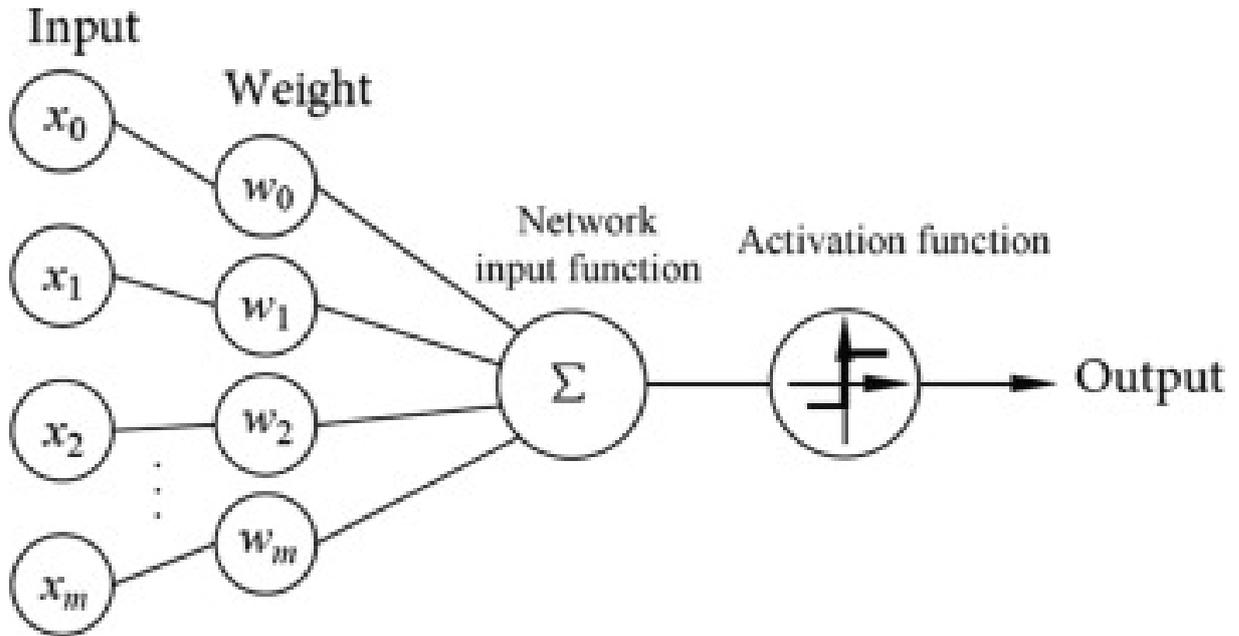
- **Neurônios Artificiais:** São as unidades básicas de processamento da rede. Cada neurônio recebe sinais de entrada, calcula uma soma ponderada (que envolve pesos e vieses) e aplica uma função de ativação para produzir uma saída.
- **Sinapses (Pesos Sinápticos):** Representam as conexões entre os neurônios. Cada sinapse possui um peso que ajusta a importância do sinal recebido, e esses pesos são atualizados durante o treinamento para que a rede aprenda as relações presentes nos dados.
- **Camadas:** As Rede Neural Artificiais (RNAs) são estruturadas em camadas. A camada de entrada recebe os dados brutos, as camadas intermediárias (ou *camadas ocultas*) processam as informações progressivamente, e a camada de saída gera o resultado final da rede.

2.2.2.1 Perceptron

O Perceptron é uma rede neural de camada única, funcionando como um classificador linear binário. Seu funcionamento matemático é fundamentado na combinação linear dos atributos de entrada e na aplicação de uma função de ativação para produzir a saída (HAYKIN, 1998).

A Figura 3 ilustra a topologia descrita a seguir:

Figura 3 – Perceptron



Fonte: Liang (2020)

De acordo com Haykin (1998), dado um vetor de entrada $\mathbf{x} = [x_1, x_2, \dots, x_n]^\top$ e um vetor de pesos $\mathbf{w} = [w_1, w_2, \dots, w_n]^\top$, o perceptron calcula a ativação a como:

$$a = \mathbf{w}^\top \mathbf{x} + b \quad (2.1)$$

Na equação 2.1, o termo b representa o viés do modelo. A saída y do perceptron é então obtida aplicando uma função de ativação $f(a)$, que pode ser descrita da seguinte forma:

$$y = f(a) = \begin{cases} 1, & \text{se } a \geq 0 \\ 0, & \text{caso contrário} \end{cases} \quad (2.2)$$

A equação 2.2 define a decisão binária do perceptron, classificando a entrada em uma das duas classes. Durante o treinamento, os pesos do perceptron são ajustados iterativamente com base no erro de classificação. Para cada exemplo de treinamento $(\mathbf{x}^{(i)}, y^{(i)})$, os pesos são atualizados segundo a seguinte regra:

$$\mathbf{w} \leftarrow \mathbf{w} + \eta(y^{(i)} - \hat{y}^{(i)})\mathbf{x}^{(i)} \quad (2.3)$$

Na equação 2.3, η é a taxa de aprendizado e $\hat{y}^{(i)}$ é a previsão atual do perceptron. Essa atualização busca minimizar o erro de classificação, ajustando os pesos para que o hiperplano de decisão se aproxime das fronteiras corretas.

Apesar de sua simplicidade, o perceptron é limitado à classificação de dados linearmente separáveis. Essa limitação motivou a criação das redes neurais multicamadas, que são capazes de lidar com não linearidades mais complexas e capturar padrões que modelos lineares não conseguem identificar (GOODFELLOW; BENGIO; COURVILLE, 2016).

2.2.3 Deep Learning

A partir das redes neurais artificiais (RNA), surgiu a demanda por modelos mais profundos e expressivos, capazes de lidar com problemas mais complexos. Essa necessidade é impulsionada por conjuntos de dados cada vez maiores e por relações temporais ou espaciais de difícil captura com técnicas tradicionais. O Deep Learning (DL) representa justamente essa evolução: um ramo do aprendizado de máquina que explora arquiteturas neurais com múltiplas camadas, capazes de extrair representações hierárquicas e não lineares dos dados. Enquanto redes simples, como o perceptron, lidam com relações básicas, o Deep Learning (DL) constrói sucessivas abstrações, permitindo que modelos aprendam características mais ricas e relevantes diretamente dos dados brutos (GOODFELLOW; BENGIO; COURVILLE, 2016).

Essas características tornam o DL capaz de resolver problemas que antes eram considerados intratáveis por métodos tradicionais. Por meio de seu funcionamento em camadas profundas, as redes de DL automatizam o processo de Representation Learning, transformando dados brutos em representações abstratas e informativas. Modelos de DL podem ser vistos como modelos de Representation Learning compostos por múltiplos níveis de representação que aprendem progressivamente funções cada vez mais complexas, de forma inteiramente automática (LECUN; BENGIO; HINTON, 2015).

É comum organizar a evolução dos modelos mais relevantes de *DL* no contexto de previsão de séries temporais de forma cronológica, partindo das primeiras arquiteturas sequenciais até alcançar as abordagens mais sofisticadas utilizadas atualmente.

2.2.3.1 Evolução dos principais modelos de Deep Learning para previsão de séries temporais

A previsão de séries temporais com DL passou por uma evolução significativa, na qual cada nova arquitetura procurou superar limitações da anterior, especialmente no que diz respeito à capacidade de capturar dependências temporais de longo alcance, lidar com múltiplas variáveis e melhorar a eficiência computacional. Esta seção apresenta os principais modelos utilizados nessa tarefa, organizados de forma cronológica.

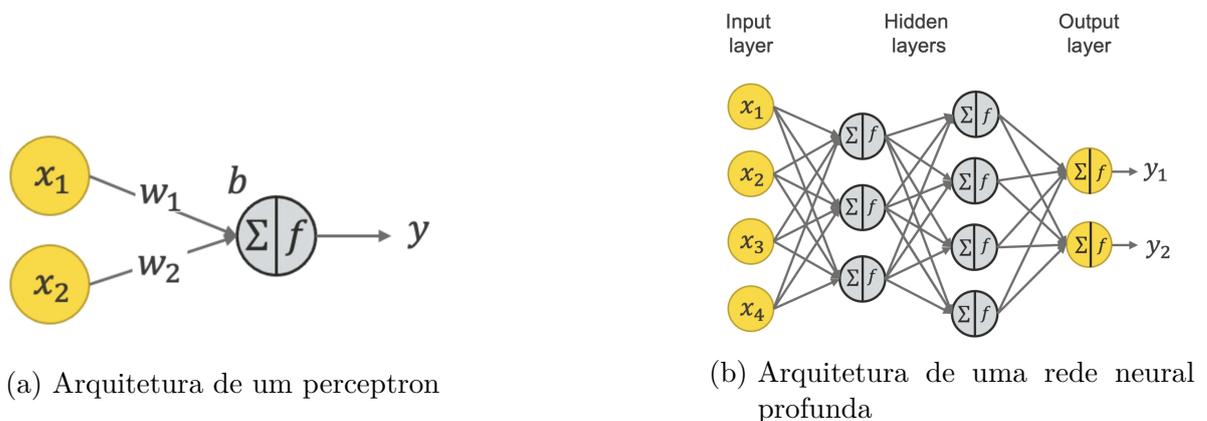
- **Redes Neurais Profundas**

As redes neurais profundas constituem a evolução natural do perceptron. Enquanto o perceptron realiza apenas uma transformação entre as entradas e a saída, exigindo

que os dados possam ser separados de forma simples, as arquiteturas profundas empilham diversas camadas de processamento. Cada nova camada refina o que a anterior já aprendeu, permitindo que o modelo descubra padrões mais sutis e relações complexas diretamente a partir dos dados. Isso amplia consideravelmente a capacidade de resolver problemas que vão além do alcance de um único perceptron (GOODFELLOW; BENGIO; COURVILLE, 2016).

A Figura 4 mostra essas diferenças. Na subfigura 4a, observa-se o perceptron com sua única camada de conexões. Já a subfigura 4b apresenta uma rede profunda feed-forward, ou seja, a informação flui em apenas uma direção: ela inicia pela camada de entrada, responsável apenas por definir a quantidade de atributos, sem realizar cálculos. Em seguida, há duas camadas ocultas em sequência: a primeira recebe os valores de entrada e gera a saída de seus três neurônios, enquanto a segunda utiliza essas saídas para calcular os quatro neurônios seguintes. Por fim, a camada de saída toma esses quatro valores como entrada e produz a predição final. Esse encadeamento entre camadas internas demonstra como a profundidade permite construir representações cada vez mais abstratas dos dados. Esse mecanismo é um dos elementos centrais dos modelos de DL (MELCHER, 2023).

Figura 4 – Comparação entre a arquitetura de um perceptron e de uma rede neural profunda



Fonte: Melcher (2023)

- **Redes Neurais Recorrentes (RNN)**

Redes Neurais Recorrentes (RNN) são redes projetadas para processar dados sequenciais, tornando-se um padrão inicial para tarefas de previsão de séries temporais, devido à sua capacidade de capturar dependências de curto prazo entre pontos consecutivos (GOODFELLOW; BENGIO; COURVILLE, 2016). Ao contrário das redes profundas feed-forward, as Rede Neural Recorrentes (RNNs) introduzem um estado oculto que se propaga de um instante para o próximo, permitindo que o

modelo retenha informações recentes. Em cada passo de tempo, o estado é combinado com a entrada corrente, gerando uma nova representação interna que reflete o histórico da sequência. No entanto, as RNNs enfrentam dificuldades para manter informações por janelas temporais longas (AMAZON WEB SERVICES, 2025).

- **Long Short-Term Memory (LSTM)**

A LSTM aperfeiçoa as RNNs ao incorporar um mecanismo de memória capaz de armazenar informações relevantes por períodos prolongados. Esse componente acumula sinais ao longo do tempo e decide quando apagar ou manter seu conteúdo, o que reduz drasticamente a perda de informação temporal. Como resultado, a LSTM é altamente eficaz na previsão de séries temporais, especialmente em janelas de previsão mais longas (LECUN; BENGIO; HINTON, 2015).

- **Transformer**

O Transformer é atualmente uma das arquiteturas mais disruptivas da inteligência artificial. Ele abriu caminho para avanços significativos em diversos campos, como processamento de linguagem natural, visão computacional e análise e previsão de séries temporais. Como foi originalmente desenvolvido para linguagem natural, a versão proposta em 2017 precisou passar por diversos ajustes para funcionar adequadamente com séries temporais, já que dados textuais e dados numéricos possuem características bastante distintas. A partir dessas adaptações, diversos modelos começaram a ser desenvolvidos com base no Transformer, incorporando melhorias para superar suas limitações e torná-lo mais eficaz na previsão de séries temporais (ZHOU et al., 2021).

Em essência, o Transformer é composto por duas partes principais: um codificador, que processa a entrada, e um decodificador, que gera a saída. Ele utiliza um mecanismo chamado autoatenção, que compara cada elemento da sequência com todos os outros e aprende quais partes “prestam atenção” umas às outras. Isso acontece de forma paralela, permitindo treinar mais rapidamente do que redes recorrentes, que leem a sequência passo a passo. Para não perder a noção de ordem, o modelo soma um código de posição em cada passo do tempo (VASWANI et al., 2017).

De acordo com Zhou et al. (2021), a aplicação direta do Transformer em tarefas de séries temporais de longo prazo apresenta três limitações principais:

- **Alto custo computacional:** o mecanismo de autoatenção tradicional possui complexidade quadrática, o que dificulta seu uso com sequências muito longas.
- **Uso excessivo de memória:** ao empilhar várias camadas com entradas longas, o modelo consome muita memória, limitando sua escalabilidade.
- **Previsão lenta:** o processo de decodificação passo a passo torna a previsão de sequências longas lenta e propensa a acúmulo de erro.

Diversos modelos posteriores, baseados na arquitetura original do Transformer, conseguiram superar essas limitações. Desde então, os Transformers têm sido bastante explorados em pesquisas voltadas à previsão de séries temporais. Novas ideias continuam surgindo e ampliando o uso dessa arquitetura nesse campo, trazendo previsões cada vez mais precisas.

2.3 MODELOS FUNDAMENTAIS DE SÉRIES TEMPORAIS (TSFM)

Modelos Fundamentais de Séries Temporais, ou *Time Series Foundation Models* (TSFM), são redes baseadas em Transformer pré-treinadas em séries temporais de múltiplos domínios (energia, clima, saúde, finanças, etc). A meta é aprender representações gerais e reutilizáveis, permitindo boa performance sem treinar um modelo do zero para cada aplicação (LIANG et al., 2024).

A Figura 5 ilustra a diferença entre os modelos tradicionais (denominados Specific na figura) e os modelos pré-treinados, mostrando que os TSFM funcionam como preditores gerais aplicáveis a diferentes cenários.

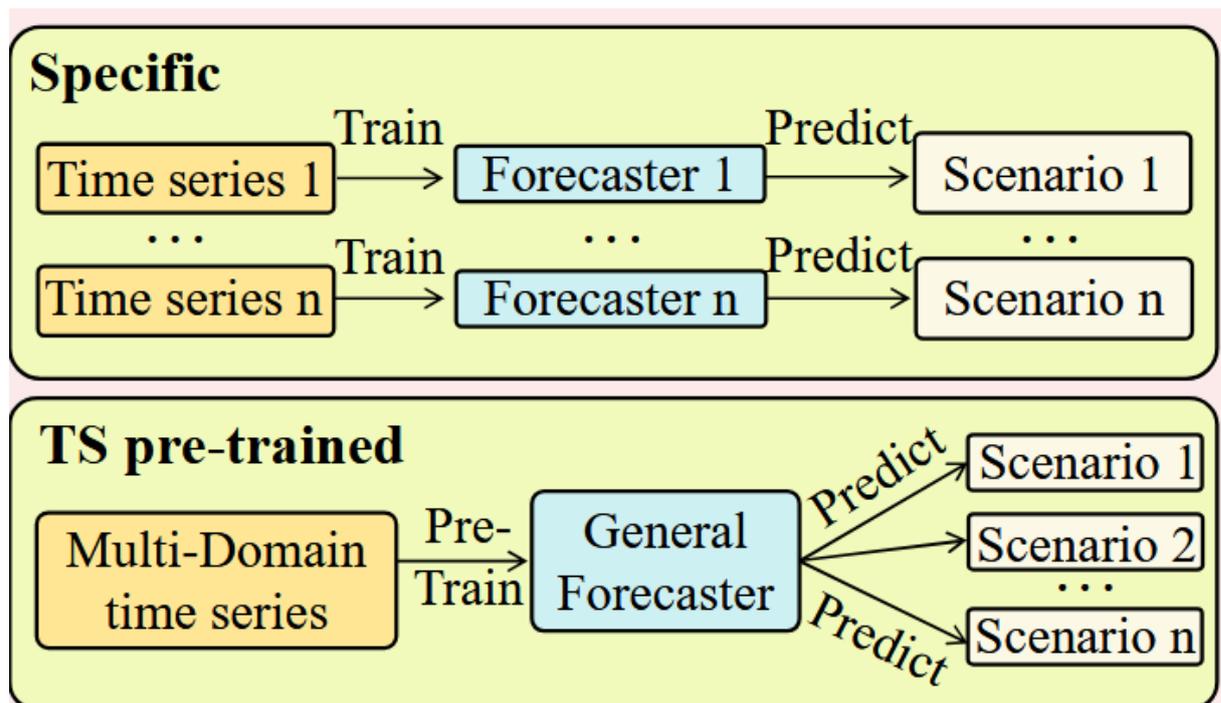


Figura 5 – Diferença entre modelos específicos e TSFM.

Fonte: Adaptado de Li et al. (2025)

Os principais TSFM têm a arquitetura baseada na estrutura do Transformer, devido à sua capacidade de capturar dependências de longo prazo em sequências temporais com alta eficiência. Esses modelos adaptam o mecanismo de atenção para lidar com a natureza contínua e multiescala dos dados temporais, sendo capazes de realizar previsões robustas mesmo em horizontes longos.

De acordo com Liang et al. (2024), uma vez que um modelo é pré-treinado, o mesmo TSFM pode ser reutilizado de três maneiras:

- **Zero-shot** – Aplicação direta a um domínio nunca visto, sem dados rotulados adicionais. Útil quando rotular é caro ou inviável.
- **Few-shot** – Ajuste leve usando um subconjunto *mínimo* de exemplos rotulados (tipicamente $\leq 5\%$ do que seria exigido num treinamento completo), suficiente para adaptar as camadas de saída ou normalizar escalas (LI et al., 2025).
- **Fine-tuning** – Re-treino parcial ou total das camadas do modelo no novo domínio.

2.4 MONASH TIME SERIES FORECASTING REPOSITORY

O *Monash Time Series Forecasting Repository* foi proposto por pesquisadores da Monash University para preencher a lacuna de um acervo abrangente que permita avaliar modelos globais ou multivariados de previsão em conjuntos heterogêneos de séries temporais. O objetivo é disponibilizar diversos conjuntos de dados acessíveis publicamente, provenientes de domínios variados e com diferenças de frequência, tamanho das séries e presença de valores ausentes, criando assim uma base padronizada para comparação de novos algoritmos de previsão (GODAHEWA et al., 2021).

Ao todo, o repositório soma trinta conjuntos de dados principais. Eles abrangem áreas como finanças, turismo, web, energia, transporte, saúde e macroeconomia, e apresentam amostragens que variam de anual até registros de alta frequência a cada quatro segundos. A partir desses trinta conjuntos, derivam-se cinquenta e oito variações, obtidas por diferenças de frequência ou pela criação de versões com e sem valores ausentes, o que amplia o leque de cenários de avaliação disponíveis (GODAHEWA et al., 2021).

Esses conjuntos de dados são referência na área de previsão de séries temporais e são amplamente utilizados na avaliação do desempenho dos principais benchmarks e modelos no estado da arte (LI et al., 2025). A padronização, diversidade de domínios e facilidade de acesso contribuem para que sejam adotados como base comum em estudos comparativos para modelos de previsão.

3 PROPOSTA

Neste capítulo, é descrita a proposta desse projeto, detalhando a metodologia de desenvolvimento que será seguida para atingir todos os objetivos previstos.

3.1 SELEÇÃO DOS CONJUNTOS DE DADOS, PRÉ-PROCESSAMENTO E ESCOLHA DOS MODELOS

Nesta etapa inicial, serão selecionados no mínimo três conjuntos de séries temporais com domínios diversos (por exemplo, demanda energética, turismo e finanças) a partir do *Monash Time Series Forecasting Repository*. Cada conjunto será submetido a um tratamento de valores faltantes, padronização de escalas e frequências, e divisão temporal em treino, validação e teste, para que todas as aplicações e os testes realizados posteriormente sejam padronizados. Além disso, será realizada a escolha de ao menos quatro TFSM open-source representativos.

3.2 PREVISÕES *ZERO-SHOT*

Os TFSM selecionados serão aplicados diretamente (*zero-shot*) aos conjuntos de dados nunca vistos durante o pré-treinamento. O desempenho será avaliado por meio das métricas MSE e MAE. Os resultados experimentais serão apresentados em tabelas, alternando os horizontes de previsão por modelo. Esses horizontes de previsão serão definidos na etapa anterior, e serão utilizados os mesmos para todos os modelos, sempre utilizando todas as métricas escolhidas.

3.3 AJUSTE FINO

Será escolhido pelo menos um dos conjuntos de dados para a realização do ajuste fino. Após a realização, diferenças de desempenho serão testadas entre o desempenho *few-shot* do modelo e sua versão re-treinada. O fine-tuning será executado em uma instância gratuita do Google Colab (GPU T4 com 16 GB).

3.4 CONSTRUÇÃO DA PLATAFORMA WEB

Será construída uma aplicação web que permite realizar previsões com os principais TFSM, selecionando de forma intuitiva os parâmetros e configurações da previsão e permitindo visualizar os resultados de forma intuitiva. A plataforma visa seguir uma ideia de ser um hub de modelos de previsão para séries temporais, seguindo uma filosofia similar à de empresas como a The Forecasting Company, que disponibiliza em uma

mesma plataforma diferentes modelos de previsão, facilitando a comparação entre eles e a escolha do modelo mais adequado para o problema em questão (THE FORECASTING COMPANY, 2025).

- (a) Upload ou seleção de séries pelo usuário;
- (b) Escolha do modelo e do horizonte de previsão;
- (c) Visualização interativa das previsões e dos valores reais, bem como métricas de erro.

A plataforma será entregue em contêiner Docker, garantindo reprodutibilidade em ambiente local e facilidade de uso.

3.5 CRONOGRAMA DE ATIVIDADES

O quadro 1 apresenta a distribuição das atividades principais ao longo de cinco meses. A parte escrita do projeto, tais como resultados, discussão, etc, será atualizada de forma contínua ao longo das atividades, conforme avançarem os experimentos.

Quadro 1 – Cronograma de atividades

Atividade	Mês				
	Ago.	Set.	Out.	Nov.	Dez.
Seleção e preparação dos dados	X				
Previsões <i>zero-shot</i>		X	X		
Ajuste fino			X	X	
Desenvolvimento da plataforma web				X	X

REFERÊNCIAS

- AMAZON WEB SERVICES. **O que é RNN? — Explicação sobre redes neurais recorrentes** — AWS. pt-BR. Acesso em: 12 jun. 2025. 2025. Disponível em: <<https://aws.amazon.com/pt/what-is/recurrent-neural-network/>>. Acesso em: 12 jun. 2025.
- BERGMANN, Dave. **What is self-supervised learning?** 5 dez. 2023. Disponível em: <<https://www.ibm.com/think/topics/self-supervised-learning>>. Acesso em: 23 jun. 2025.
- CHEN, Si-An et al. **TSMixer: An All-MLP Architecture for Time Series Forecasting**. 2023. arXiv: 2303.06053 [cs.LG]. Disponível em: <<https://arxiv.org/abs/2303.06053>>.
- DAS, Abhimanyu et al. **A decoder-only foundation model for time-series forecasting**. 2024. arXiv: 2310.10688 [cs.CL]. Disponível em: <<https://arxiv.org/abs/2310.10688>>.
- DEVOPEDIA. **Supervised vs Unsupervised Learning**. Version 33, January 12. Accessed 2025-06-08. 2022. Disponível em: <<https://devopedia.org/supervised-vs-unsupervised-learning>>.
- GODAHEWA, Ruvinda et al. Monash Time Series Forecasting Archive. In: NEURIPS Datasets and Benchmarks Track. 2021. Disponível em: <<https://forecastingdata.org/>>.
- GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep Learning**. MIT Press, 2016. <http://www.deeplearningbook.org>.
- HAYKIN, Simon. **Neural Networks: A Comprehensive Foundation**. 2nd. USA: Prentice Hall PTR, 1998. ISBN 0132733501.
- LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. eng. **Nature**, v. 521, n. 7553, p. 436–444, mai. 2015. ISSN 1476-4687. DOI: 10.1038/nature14539.
- LI, Zhe et al. **TFSM-Bench: A Comprehensive and Unified Benchmark of Foundation Models for Time Series Forecasting**. arXiv, jun. 2025. arXiv:2410.11802 [cs]. DOI: 10.48550/arXiv.2410.11802. Disponível em: <<http://arxiv.org/abs/2410.11802>>. Acesso em: 19 jun. 2025.
- LIANG, Xiaoyao. Chapter 1 - Theoretical basis. In _____. **Ascend AI Processor Architecture and Programming**. Edição: Xiaoyao Liang. Elsevier, jan. 2020. P. 1–40. ISBN 978-0-12-823488-4. DOI: 10.1016/B978-0-12-823488-4.00001-1. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780128234884000011>>. Acesso em: 9 jun. 2025.

LIANG, Yuxuan et al. Foundation Models for Time Series Analysis: A Tutorial and Survey. In: PROCEEDINGS of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. ACM, ago. 2024. (KDD '24), p. 6555–6565. DOI: 10.1145/3637528.3671451. Disponível em: <<http://dx.doi.org/10.1145/3637528.3671451>>.

MELCHER, Kathrin. **A Friendly Introduction to Deep Neural Networks**. Blog post. KNIME. 2023. Disponível em: <<https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks>>. Acesso em: 9 jun. 2025.

TIME Series Data Introduction. en-US. Accessed 2025-06-08. MongoDB. [s.d.] Disponível em: <<https://www.mongodb.com/resources/basics/time-series-data-analysis>>.

MONTGOMERY, Douglas C.; JENNINGS, Cheryl L.; KULAHCI, Murat. **Introduction to Time Series Analysis and Forecasting**. 2. ed. Hoboken, NJ: John Wiley & Sons, 2015. P. 672. (Wiley Series in Probability and Statistics). ISBN 978-1-118-74511-3.

MURPHY, Kevin P. **Machine Learning: A Probabilistic Perspective**. The MIT Press, 2012. ISBN 0262018020.

REIS, Marcelo Menezes. **Análise de séries temporais**. 2023. <https://www.inf.ufsc.br/~marcelo.menezes.reis/Cap4.pdf>. Florianópolis, p. 55. Acesso em: 11 maio 2025.

THE FORECASTING COMPANY. **Forecast anything with high accuracy, with world context, in seconds**. Accessed 8 July 2025. The Forecasting Company. 2025. Disponível em: <<https://www.theforecastingcompany.com/en/>>. Acesso em: 8 jul. 2025.

VASWANI, Ashish et al. Attention is All you Need. In: ADVANCES in Neural Information Processing Systems. Curran Associates, Inc., 2017. v. 30. Disponível em: <https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html>. Acesso em: 18 jun. 2025.

ZHOU, Haoyi et al. **Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting**. arXiv, mar. 2021. arXiv:2012.07436 [cs]. DOI: 10.48550/arXiv.2012.07436. Disponível em: <<http://arxiv.org/abs/2012.07436>>. Acesso em: 18 jun. 2025.