# UNIVERSIDADE ESTADUAL PAULISTA - UNESP

Faculdade de Engenharia (FEIS)- campus de Ilha Solteira

**DANIELI BIAGI VILELA**


**Aprendizado por transferência utilizando a rede neural ARTMAP Fuzzy.**


Ilha Solteira

2026

**Danieli Biagi Vilela**

# Aprendizado por transferência utilizando a rede neural ARTMAP Fuzzy.

Tese, apresentada à Universidade Estadual Paulista (UNESP), Faculdade de Engenharia (FEIS), Ilha Solteira, para obtenção do título de Doutora em Engenharia Elétrica.

Área de Concentração: Automação

Orientador: Profª Dra. Anna Diva Plasencia Lotufo

Ilha Solteira

2026

FICHA CATALOGRÁFICA
Desenvolvida pela Diretoria Técnica de Biblioteca e Documentação

Vilela, Danieli Biagi.
V699a Aprendizado por transferência utilizando a rede neural ARTMAP Fuzzy / Danieli Biagi Vilela. -- Ilha Solteira: [s.n.], 2026
87 f. : il.

Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2026

Orientadora: Anna Diva Plasencia Lotufo

Inclui bibliografia

1. Aprendizado por transferência. 2. ARTMAP Fuzzy. 3. Adaptação de domínio. 4. Dados de sensores. 5. Detecção de ocupação.

Elaborada por Raiane da Silva Santos - CRB-8/9999

UNIVERSIDADE ESTADUAL PAULISTA

**Câmpus de Ilha Solteira**

# CERTIFICADO DE APROVAÇÃO

TÍTULO DA TESE: Aprendizado por transferência utilizando a rede neural Fuzzy ARTMAP

AUTORA: DANIELI BIAGI VILELA
ORIENTADORA: ANNA DIVA PLASENCIA LOTUFO

Aprovada como parte das exigências para obtenção do Título de Doutora em Engenharia Elétrica, área: Automação pela Comissão Examinadora:

gov.br Documento assinado digitalmente
ANNA DIVA PLASENCIA LOTUFO
Data: 19/02/2026 14:27:04-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. ANNA DIVA PLASENCIA LOTUFO (Participaçao Presencial)
Departamento de Engenharia Elétrica / UNESP / Câmpus de Ilha Solteira - FEIS

gov.br Documento assinado digitalmente
MARA LUCIA MARTINS LOPES
Data: 16/02/2026 11:24:16-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. MARA LUCIA MARTINS LOPES (Participaçao Presencial)
Departamento de Matemática / UNESP / Câmpus de Ilha Solteira - FEIS

gov.br Documento assinado digitalmente
THAYS APARECIDA DE ABREU SANTOS
Data: 19/02/2026 14:20:11-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. THAYS APARECIDA DE ABREU SANTOS (Participaçao Presencial)
Departamento de Matemática / UNESP / Câmpus de Ilha Solteira - FEIS

gov.br Documento assinado digitalmente
KENJI NOSE FILHO
Data: 12/02/2026 19:51:26-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. KENJI NOSE FILHO (Participaçao Virtual)
Centro de Engenharia, Modelagem e Ciências Sociais Aplicadas / UFABC / Universidade Federal do ABC

gov.br Documento assinado digitalmente
NAYARA LONGO SARTOR
Data: 06/02/2026 19:23:51-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. NAYARA LONGO SARTOR ZAGUI (Participaçao Virtual)
IFMT / Instituto Federal de Educação, Ciência e Tecnologia de Mato Grosso

Ilha Solteira, 06 de fevereiro de 2026.

## IMPACTO POTENCIAL DESTA PESQUISA

Esta pesquisa apresenta potencial para contribuir de forma incremental para o avanço do conhecimento científico nesta linha de investigação, em contextos local e internacional, por meio da proposição, adaptação, experimentação e análise da aplicação de uma técnica de aprendizado de máquina, especificamente o aprendizado por transferência, à rede neural artificial Fuzzy ARTMAP. De acordo com o levantamento bibliográfico realizado, tal aplicação configura-se como inédita na literatura especializada. Adicionalmente, o estudo poderá contribuir para a discussão sobre o desenvolvimento sustentável de algoritmos, na medida em que a utilização de modelos previamente treinados em cenários correlatos poderá reduzir demandas computacionais, custos financeiros, esforço humano e tempo de desenvolvimento, bem como atenuar limitações associadas à indisponibilidade de dados compatíveis.

## POTENTIAL IMPACT OF THIS RESEARCH

This research has the potential to contribute incrementally to the advancement of scientific knowledge in this line of inquiry, both locally and internationally, through the proposal, adaptation, experimentation, and analysis of the application of a machine learning technique, specifically transfer learning, to the Fuzzy ARTMAP artificial neural network. According to the bibliographic survey conducted, such an application is unprecedented in the specialized literature. Furthermore, the study may contribute to the discussion on sustainable algorithm development, insofar as the use of models previously trained in related scenarios may reduce computational demands, financial costs, human effort, and development time, while also mitigating limitations associated with the unavailability of compatible data.

## AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), pela concessão de bolsa de pesquisa (processo 302948/2022-8).

A Universidade Estadual Paulista "Júlio de Mesquita Filho", a Faculdade de Engenharia - FEIS e ao Laboratório de Sistemas Inteligentes (SINTEL).

A orientadora Profª. Dra. Anna Diva Plasencia Lotufo.

## RESUMO

A maioria dos algoritmos de aprendizado de máquina são utilizados sob a suposição de que os dados de treinamento e teste possuem a mesma distribuição estatística. Essa suposição pode nem sempre estar correta, principalmente porque muitos tipos de dados se tornam desatualizados rapidamente. O aprendizado por transferência tem por objetivo utilizar o conhecimento adquirido em um domínio de origem para melhorar o aprendizado de um domínio de destino relacionado, sob a suposição de que se são realmente relacionados, pode-se descobrir quais partes do conhecimento extraído de um domínio podem ser utilizadas para treinar outro. Neste trabalho foi testado o potencial da rede ARTMAP *Fuzzy* (FAM) para generalizar o conhecimento e executar transferência estrutural, a partir de diferentes estratégias de transferências (diretas e seletivas). Foi transferido conhecimento de um domínio de origem para dois alvos, um com dados reais e outro sintético gerado a partir do deslocamento do primeiro. O domínio de origem é um banco de dados para contagem de ocupação de um ambiente e o alvo real é um banco de dados para detecção da ocupação, ambos obtidos a partir de dados de sensores, com vários tipos de variáveis ambientais. Com os resultados experimentais, foi possível concluir que, na metodologia proposta, a rede FAM executa adequadamente a transferência e mantêm a robustez no aprendizado frente a diferentes intensidades de deslocamento, elevando consideravelmente o número de categorias criadas para agrupar os protótipos dos domínios alvo. E, por meio dos diferentes tipos de estratégia de transferência testadas, foi possível obervar que estratégias seletivas em relação a matriz $\mathbf{W}^a$ são mais vantajosas, considerando todos os critérios de análise de resultados analisados.



**Palavras-Chave:** aprendizado por transferência; ARTMAP *Fuzzy*; adaptação de domínio; dados de sensores; detecção de ocupação.

## ABSTRACT

Most machine learning algorithms are commonly applied under the assumption that the training and test data follow the same statistical distribution. This assumption may not always hold, particularly because many types of data become outdated rapidly. Transfer learning aims to leverage knowledge acquired in a source domain to improve learning in a related target domain, under the premise that, if the domains are indeed related, it is possible to identify which parts of the knowledge extracted from one domain can be reused to train another. In this work, we investigated the potential of the Fuzzy ARTMAP (FAM) neural network to generalize knowledge and perform structural transfer through different transfer strategies (direct and selective). Knowledge was transferred from a source domain to two targets: one containing real data and another synthetic target generated by shifting the real target. The source domain comprises a dataset for occupancy counting in an environment, and the real target comprises a dataset for occupancy detection; both were obtained from sensor data with multiple types of environmental variables. Experimental results indicate that, within the proposed methodology, FAM performs transfer effectively and remains robust under different magnitudes of distribution shift, albeit with a considerable increase in the number of categories created to cluster prototypes in the target domains. Moreover, among the transfer strategies evaluated, selective strategies with respect to the matrix $\mathbf{W}^a$ were found to be more advantageous according to all analysis criteria considered.



**Keywords:** transfer-learning; Fuzzy ARTMAP; domain-adaptation; sensor data; occupancy detection.

# LISTA DE ILUSTRAÇÕES


Quadro 1 Cenários de transferência de $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ na rede FAM e comportamento esperado. . . . . 45
Figura 1 Visão geral do método: $S \rightarrow$ treino $\rightarrow$ transferência $\rightarrow \{T_1, T_2(\lambda)\}$. . . . 48
Quadro 2 Variáveis do banco de dados *Occupancy Detection*. . . . . 49
Quadro 3 Variáveis do banco de dados *Room Occupancy Estimation*. . . . . 50
Figura 2 Resumo (frac = 1): $\Delta$Acc em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia. . 65
Figura 3 Resumo (frac = 1): $\Delta F_1$ em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia. . . 66
Figura 4 Resumo (frac = 1): $\Delta$AUC-ROC em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia. . . . . 66
Figura 5 Resumo (frac = 1): número de categorias ao longo de $\lambda$, por estratégia. . . 67
Figura 6 Curvas de aprendizado em $\mathbf{T}_1$: acurácia em função da fração de dados rotulados (`frac`). . . . . 68
Figura 7 Curvas de aprendizado em $\mathbf{T}_1$: $F_1$ em função da fração de dados rotulados (`frac`). . . . . 68
Figura 8 Curvas de aprendizado em $\mathbf{T}_1$: AUC-ROC em função da fração de dados rotulados (`frac`). . . . . 69
Figura 9 Curvas de aprendizado em $\mathbf{T}_1$: número de categorias em função da fração de dados rotulados (`frac`). . . . . 70

## LISTA DE TABELAS


Tabela 1 – Diferenças entre as configurações de AT em relação à existência de rótulos . 27
Tabela 2 – Classificações clássicas de AT: cenário, hipótese principal e estratégias típicas. 28
Tabela 3 – Famílias de modelos/arquiteturas frequentemente usados em AT e exemplos. 30
Tabela 4 – Parâmetros fixados para a geração do *shift* em $\mathbf{T}_2(\lambda)$. . . . . 55
Tabela 5 – Grade de valores utilizada na análise de sensibilidade (*grid search*) dos hiperparâmetros e do nível de *shift*. . . . . 61
Tabela 6 – Métricas de distância entre domínios ($S, T_1$ e $T_2(\lambda)$) em função de $\lambda$. . . . . 63
Tabela 7 – Desempenho em $\mathbf{T}_1$ (fração $= 1,0$). Média $\pm$ desvio padrão. . . . . 65
Tabela 8 – Variação de acurácia (em pontos percentuais) relativa ao TFS no regime fração $= 1,0$. . . . . 69
Tabela 9 – Número médio de categorias (*cat*) no regime fração $= 1,0$. . . . . 70
Tabela 10 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 0{,}25$): acurácia média (±DP) por fração de dados-alvo. . . . . 71
Tabela 11 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 0{,}25$): F1 médio (±DP) por fração de dados-alvo. . . . . 71
Tabela 12 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 1{,}00$): acurácia média (±DP) por fração de dados-alvo. . . . . 71
Tabela 13 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 1{,}00$): F1 médio (±DP) por fração de dados-alvo. . . . . 72
Tabela 14 – Resultados da transferência seletiva (Top-$\mathbf{k}$) variando o número de categorias transferidas. . . . . 73
Tabela 15 – Melhores combinações de $\rho$ e $\beta$. . . . . 74
Tabela 16 – Sensibilidade da acurácia à variação de $\rho$ e $\beta$. . . . . 75

## LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| ADDA | Adaptação adversarial discriminativa de domínio |
| AM | Aprendizado de máquina |
| ART | Teoria da Ressonância Adaptativa |
| ARTMAP | *Adaptive Resonance Theory Mapping* |
| AT | Aprendizado por transferência |
| $\text{PRUNED}_{\text{Wa}}$ | Poda + readaptação restrita ($\mathbf{W}^a$) |
| $\text{PRUNED}_{\text{WaWab}}$ | Poda + readaptação ampliada ($\mathbf{W}^a$ + $\mathbf{W}^{ab}$) |
| $\text{TOPK}_{\text{Wa}}$ | Seleção Top-$\mathbf{K}$ + readaptação restrita ($\mathbf{W}^a$) |
| $\text{TOPK}_{\text{Wab}}$ | Seleção Top-$\mathbf{K}$ + readaptação ampliada ($\mathbf{W}^a$ + $\mathbf{W}^{ab}$) |
| $\text{AT}_{\text{Wa}}$ | Atualização restrita da matriz $\mathbf{W}^a$ ($\mathbf{W}^a$) |
| $\text{AT}_{\text{Wab}}$ | Atualização ampliada das matrizes $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ ($\mathbf{W}^a$ + $\mathbf{W}^{ab}$) |
| AP | Aprendizado profundo |
| AR | Aprendizado por reforço |
| AUC | *Area Under the Curve* - Área sob a curva |
| AUC-ROC | Área sob a curva ROC |
| CNN | *Convolutional neural networks* |
| CORAL | *Correlation Alignment* - Alinhamento de correlação |
| DANN | *Domain-Adversarial Neural Network* - Rede neural adversarial ao domínio |
| DP | Desvio padrão |
| ERM | *Empirical Risk Minimization* - Minimização do risco empírico |
| F1 | *F1-score* – medida F1 |
| FAM | *Fuzzy* ARTMAP |
| | GNNs (*Graph Neural Networks* ou Redes Neurais de Grafos) |
| LSTM | (*Long short-term memory* ou memória de longo e curto prazo) |

| | |
|---|---|
| MMD | *Maximum Mean Discrepancy* - Discrepância de médias máximas |
| MMD-RBF | MMD com kernel RBF (*Radial Basis Function*) |
| MTL | *Multi-task learning* |
| PAD | *Proxy A-Distance* |
| PIR | *Passive Infrared* - Sensor infravermelho passivo |
| RBF | *Radial Basis Function* |
| RKHS | *Reproducing Kernel Hilbert Space* |
| RNA | Rede neural artificial |
| ROC | *Receiver Operating Characteristic* |
| TFS | *Train From Scratch* - Treinamento do zero |
| TL | *Transfer Learning* (Aprendizado por transferência) |
| TL_Pruned | Poda por pureza (*pruning*) |
| $\text{PRUNED}_{\text{Wa}}$ | Poda + readaptação restrita ($\mathbf{W}^a$) |
| $\text{PRUNED}_{\text{WaWab}}$ | Poda + readaptação ampliada ($\mathbf{W}^a$ e $\mathbf{W}^{ab}$) |
| $\text{TOPK}_{\text{Wa}}$ | Seleção Top-$\mathbf{K}$ + readaptação restrita ($\mathbf{W}^a$) |
| $\text{TOPK}_{\text{Wab}}$ | Seleção Top-$\mathbf{K}$ + readaptação ampliada ($\mathbf{W}^a$ e $\mathbf{W}^{ab}$) |
| $\text{AT}_{\text{Wa}}$ | Adaptação restrita (atualização apenas de $\mathbf{W}^a$) |
| $\text{AT}_{\text{Wab}}$ | Adaptação ampliada (atualização de $\mathbf{W}^a$ e $\mathbf{W}^{ab}$) |
| Top-$\mathbf{K}$ | Transferência seletiva (*Top*-$\mathbf{K}$) |
| UCI | *University of California, Irvine* (UCI Machine Learning Repository) |
| ZS | *Zero-Shot* |

## LISTA DE SÍMBOLOS

**Aprendizado por transferência: domínios, tarefas e dados**

| | |
|---|---|
| $\mathcal{D}$ | Domínio, definido por $\mathcal{D} = \{\mathcal{X}, P(X)\}$ |
| $\mathcal{D}_\mathcal{S}$ | Domínio-fonte (*source domain*) |
| $\mathcal{D}_\mathcal{T}$ | Domínio-alvo (*target domain*) |
| $\mathcal{T}$ | Tarefa, definida por $\mathcal{T} = \{\mathcal{Y}, f(.)\}$ |
| $\mathcal{T}_\mathcal{S}$ | Tarefa-fonte (*source task*) |
| $\mathcal{T}_\mathcal{T}$ | Tarefa-alvo (*target task*) |
| $\mathcal{X}$ | Espaço de atributos (espaço de entrada) |
| $\mathcal{Y}$ | Espaço de rótulos (espaço de saída) |
| $\mathcal{X}_S$, $\mathcal{X}_T$ | Espaços de atributos em fonte e alvo, respectivamente |
| $\mathcal{Y}_S$, $\mathcal{Y}_T$ | Espaços de rótulos em fonte e alvo, respectivamente |
| $D_S$ | Conjunto de dados do domínio-fonte (pares supervisionados) |
| $D_T$ | Conjunto de dados do domínio-alvo (pares supervisionados) |
| $n$ | Número de amostras no conjunto, quando não se distingue fonte e alvo |
| $n_S$ | Número de amostras no domínio-fonte |
| $n_T$ | Número de amostras no domínio-alvo |
| $X$ | Variável aleatória associada às entradas |
| $Y$ | Variável aleatória associada aos rótulos/saídas |
| $\mathbf{x}$ | Realização (amostra) no espaço de entrada, com $\mathbf{x} \in \mathcal{X}$ |
| $y$ | Realização (rótulo) no espaço de saída, com $y \in \mathcal{Y}$ |
| $(\mathbf{x}_i, y_i)$ | Par supervisionado (amostra e rótulo) |
| $i$ | Índice de amostra |
| $P(X)$ | Distribuição marginal das entradas no domínio considerado |
| $P(Y \mid X)$ | Distribuição condicional (relação rótulo–entrada) |

| | |
|---|---|
| $P_S(X_S),\ P_T(X_T)$ | Distribuições marginais em fonte e alvo |
| $P_S(Y_S \mid X_S),\ P_T(Y_T \mid X_T)$ | Distribuições condicionais em fonte e alvo |
| $\mathbf{S}$ | Amostra/distribuição do domínio-fonte (notação teórica) |
| $\mathbf{T}$ | Amostra/distribuição do domínio-alvo (notação teórica) |
| | **Hipóteses, perdas e limites teóricos** |
| $f(.)$ | Função preditiva (modelo/hipótese) |
| $f_T(.)$ | Função preditiva no domínio-alvo |
| $\hat{f}$ | Preditor estimado por minimização de risco empírico |
| $\mathcal{F}$ | Conforme o contexto: (i) classe de hipóteses; (ii) conjunto de atributos comuns após harmonização |
| $\mathcal{L}(\cdot, \cdot)$ | Função de perda |
| $h$ | Classificador/hipótese em uma classe $\mathcal{H}$ |
| $\epsilon_S(h)$ | Risco/erro de $h$ no domínio-fonte |
| $\epsilon_T(h)$ | Risco/erro de $h$ no domínio-alvo |
| $d_{\mathcal{H}\Delta\mathcal{H}}(S, T)$ | Divergência $\mathcal{H}\Delta\mathcal{H}$ entre fonte e alvo |
| $\mathcal{H}\Delta\mathcal{H}$ | Classe pela diferença simétrica de hipóteses em $\mathcal{H}$ |
| $\Delta$ | Operador de diferença simétrica (na construção $\mathcal{H}\Delta\mathcal{H}$) |
| $\lambda$ | Conforme o contexto: (i) intensidade de *shift* em $\mathbf{T}_2(\lambda)$; (ii) termo de incompatibilidade no limite teórico |
| | **Síntese de domínio-alvo e *shift*** |
| $\mathbf{T}_1$ | Domínio-alvo real (dados-alvo observados) |
| $\mathbf{T}_1^{\text{shift}}$ | Versão deslocada de $\mathbf{T}_1$ usada na síntese de $\mathbf{T}_2(\lambda)$ |
| $\mathbf{T}_2(\lambda)$ | Domínio-alvo sintético obtido de $\mathbf{T}_1$ com deslocamento controlado por $\lambda$ |
| $\mathbf{x}_{T_1}$ | Vetor de atributos (uma instância) no domínio $\mathbf{T}_1$ |
| $\mathbf{x}_{T_1^{\text{shift}}}$ | Vetor de atributos após aplicar *shift* em $\mathbf{T}_1$ |
| $\mathbf{x}_{T_2(\lambda)}$ | Vetor de atributos no domínio sintético $\mathbf{T}_2(\lambda)$ |

| | |
|---|---|
| $x'$ | Atributo após normalização min–max |
| $\tilde{x}$ | Atributo após transformação paramétrica de *shift* |
| $\mu$ | Média estimada de um atributo no domínio considerado |
| $g$ | Termo de escala (*gain*) |
| $\sigma$ | Desvio-padrão do ruído gaussiano |
| $\varepsilon$ | Ruído aleatório; tipicamente $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ |
| $\mathcal{N}(0, \sigma^2)$ | Distribuição normal (gaussiana) |
| $\mathrm{clip}(\cdot)$ | Operador que limita o valor ao intervalo especificado (por exemplo, $[0, 1]$) |
| $\mathcal{G}_T$, $\mathcal{G}_L$, $\mathcal{G}_C$ | Grupos de atributos: temperatura, luminosidade e $\mathrm{CO}_2$ |
| $\mathrm{CO}_2$ | Concentração medida de dióxido de carbono |
| $\mathcal{F}_S$ | Conjunto de atributos disponíveis no domínio-fonte |
| $\mathcal{F}_{T_1}$ | Conjunto de atributos disponíveis em $\mathbf{T}_1$ |
| $d$ | Dimensionalidade do espaço de atributos após pré-processamento/harmonização |
| $\mathbb{R}$ | Conjunto dos números reais |
| $\mathbf{X}_S$ | Matriz de atributos do domínio-fonte após harmonização |
| $\mathbf{X}_{T_1}$ | Matriz de atributos do domínio-alvo $\mathbf{T}_1$ após harmonização |
| | **Medição de distâncias entre domínios** |
| $z$ | Rótulo artificial de domínio, com $z = 0$ para amostras de $\mathbf{S}$ e $z = 1$ para amostras de $\mathbf{T}$ |
| $g_{S/T}$ | Classificador binário de domínio treinado para distinguir $\mathbf{S}$ de $\mathbf{T}$ |
| $\hat{\epsilon}_{S/T}$ | Erro empírico do classificador de domínio $g_{S/T}$ |
| $\mathrm{Acc}_{S/T}$ | Acurácia do classificador de domínio, com $\mathrm{Acc}_{S/T} = 1 - \hat{\epsilon}_{S/T}$ |
| $d_{\mathcal{A}}(S, T)$ | Divergência $\mathcal{A}$ entre $\mathbf{S}$ e $\mathbf{T}$, aproximada via PAD |
| $\mathrm{PAD}(S, T)$ | *Proxy A-Distance* entre $\mathbf{S}$ e $\mathbf{T}$ |
| $k(\cdot, \cdot)$ | Função *kernel* positiva definida |
| $\phi(\cdot)$ | Mapeamento de características induzido por $k$ |

| | |
|---|---|
| $\mathcal{H}_k$ | Espaço de Hilbert reproduzível (RKHS) associado ao *kernel* $k$ |
| $\|\cdot\|_{\mathcal{H}_k}$ | Norma no espaço $\mathcal{H}_k$ |
| $\mathrm{MMD}(S, T)$ | *Maximum Mean Discrepancy* entre $\mathbf{S}$ e $\mathbf{T}$ |
| $\mathbf{C}_S$, $\mathbf{C}_T$ | Matrizes de covariância amostral das *features* em $\mathbf{S}$ e $\mathbf{T}$ |
| $\mathrm{CORAL}(S, T)$ | *Correlation Alignment* entre $\mathbf{S}$ e $\mathbf{T}$ |
| | **Rede ARTMAP Fuzzy (FAM)** |
| $\mathrm{ART}_a$ | Módulo ART associado ao espaço de entrada (atributos) |
| $\mathrm{ART}_b$ | Módulo ART associado ao espaço de rótulos/saídas |
| $\alpha$ | Parâmetro de escolha da função de ativação competitiva $T_j(\mathbf{A})$ |
| $\beta$ | Taxa de aprendizado, conforme o contexto: $\mathrm{ART}_a$ ou hiperparâmetro global |
| $\beta_{ab}$ | Taxa de aprendizado do campo de mapeamento (pesos $\mathbf{w}_j^{ab}$) |
| $\rho$ | Parâmetro de vigilância, na forma agregada quando não se distingue módulo |
| $\rho_a$ | Parâmetro de vigilância do módulo $\mathrm{ART}_a$ |
| $\rho_{ab}$ | Parâmetro de vigilância do campo de mapeamento (consistência supervisionada) |
| $\epsilon$ | Incremento positivo usado no *match tracking* para elevar $\rho_a$ |
| $\rho^\star$, $\beta^\star$ | Valores selecionados por *tuning* (melhores hiperparâmetros) |
| $M$ | Dimensão do vetor de entrada original $\mathbf{a}$, antes da codificação complementar |
| $2M$ | Dimensão do vetor $\mathbf{A}$ após codificação complementar |
| $K$ | Conforme o contexto: (i) número de classes/rótulos (dimensão de $\mathbf{b}$); (ii) número de categorias selecionadas na estratégia *Top-K* |
| $C$ | Número de categorias selecionadas (*committed*) no módulo $\mathrm{ART}_a$ |
| $N_{\mathrm{cat}}$ | Número de categorias existentes no modelo (camada competitiva) |
| $\mathbf{a}$ | Vetor de entrada original, com $\mathbf{a} \in [0, 1]^M$ |
| $\mathbf{a}^c$ | Complemento de $\mathbf{a}$, com $\mathbf{a}^c = \mathbf{1} - \mathbf{a}$ |
| $\mathbf{A}$ | Vetor de entrada com codificação complementar, $\mathbf{A} = (\mathbf{a}, \mathbf{a}^c) \in [0, 1]^{2M}$ |

| | |
|---|---|
| $\mathbf{b}$ | Vetor de rótulo/saída (tipicamente *one-hot*), com $\mathbf{b} \in [0,1]^K$ |
| $\mathbf{1}$ | Vetor de uns, com dimensão compatível ao contexto |
| $\mathbf{w}_j^a$ | Vetor de pesos (protótipo) da categoria $j$ no módulo $\mathrm{ART}_a$ |
| $\mathbf{w}_j^{ab}$ | Vetor de pesos do campo de mapeamento associado à categoria $j$ |
| $\mathbf{W}^a$ | Matriz de protótipos; linhas $(\mathbf{w}_j^a)^\top$, com $\mathbf{W}^a \in [0,1]^{C\times 2M}$ |
| $\mathbf{W}^{ab}$ | Matriz de mapeamento; linhas $(\mathbf{w}_j^{ab})^\top$, com $\mathbf{W}^{ab} \in [0,1]^{C\times K}$ |
| $\mathbf{w}_{J,k}^{ab}$ | Componente $k$ do vetor $\mathbf{w}_J^{ab}$, usada na decisão de classe |
| $T_j(\mathbf{A})$ | Função de escolha (ativação competitiva) da categoria $j$ |
| $\wedge$ | Operador *fuzzy AND* (mínimo componente a componente) |
| $\|\cdot\|_1$ | Norma $L_1$ (soma dos módulos das componentes) |
| $\|\mathbf{x}\|$ | Abuso notacional para $\|\mathbf{x}\|_1$, isto é, $\|\mathbf{x}\| = \sum_i x_i$ |
| $\|\cdot\|_F$ | Norma de Frobenius |
| $\top$ | Transposição, por exemplo $(\mathbf{w}_j)^\top$ |
| | **Estratégias *Top-K* e poda (*pruning*)** |
| $r_j$ | Relevância (massa $L_1$) do protótipo da categoria $j$ |
| $w_{j,p}$ | Componente $p$ do protótipo $\mathbf{w}_j$ |
| $p$ | Índice de componente de vetor |
| $K_{\text{eff}}$ | Valor efetivo de $K$ limitado ao intervalo $[1,\ N_{\text{cat}}]$ |
| $\mathcal{C}_K$ | Conjunto das $K$ categorias selecionadas para transferência seletiva |
| $\mathcal{K}$ | Conjunto de índices das categorias selecionadas por *TopK* |
| $\mathcal{I}_c$ | Índices das amostras do alvo cuja categoria vencedora é $c$ |
| $n_{c,k}$ | Número de amostras em $\mathcal{I}_c$ pertencentes à classe $k$ |
| $\mathrm{purity}(c)$ | Pureza da categoria $c$, com $\mathrm{purity}(c) \in [0,1]$ |
| $\tau$ | Limiar de pureza para poda de categorias |
| $j$ | Índice de categoria (FAM) ou índice genérico, conforme o contexto |
| $J$ | Índice da categoria vencedora selecionada por competição/ressonância (FAM) |

| | |
|---|---|
| $k$ | Índice de classe (componente do vetor de rótulos) |
| $\hat{k}$ | Classe predita (índice estimado) |
| | **Métricas de avaliação** |
| $\mathrm{Acc}$ | Acurácia |
| $\mathrm{I}(\cdot)$ | Função indicadora (vale 1 se a condição é verdadeira, e 0 caso contrário) |
| $\hat{y}_i$ | Rótulo predito para a $i$-ésima amostra |
| $\mathrm{Prec}_k$ | Precisão da classe $k$ |
| $\mathrm{Rec}_k$ | Revocação da classe $k$ |
| $\mathrm{F1}_k$ | F1-score da classe $k$ |
| $\mathrm{F1}_{\text{macro}}$ | F1-score macro (média sobre classes) |
| $\Delta\mathrm{Acc}$ | Diferença de acurácia em relação a um *baseline* |
| $\Delta\mathrm{F1}$ | Diferença de F1-score em relação a um *baseline* |

**SUMÁRIO**


**1 INTRODUÇÃO . . . . . 20**

**2 APRENDIZADO POR TRANSFERÊNCIA . . . . . 22**
2.1 Diferenças entre o AT e o AM tradicional . . . . . 22
2.2 Breve histórico e primeiros trabalhos . . . . . 24
2.3 Conceitos fundamentais e notação clássica . . . . . 25
**2.3.1 Classificação das técnicas de aprendizado por transferência . . . . . 26**
2.4 Principais modelos e arquiteturas utilizados na literatura de AT . . . . . 29
**2.4.1 Modelos "clássicos" (pré-aprendizado profundo): reponderação, expansão de atributos e correspondências . . . . . 30**
**2.4.2 Redes neurais profundas para visão computacional: CNNs, atributos profundos e *fine-tuning* . . . . . 31**
**2.4.3 Pré-treinamento auto-supervisionado e contrastivo: representações "gerais" para múltiplos alvos . . . . . 31**
**2.4.4 Transformers em NLP . . . . . 31**
**2.4.5 Transformers em visão computacional . . . . . 32**
**2.4.6 Adaptação de domínio com aprendizado profundo: alinhamento estatístico adversarial . . . . . 32**
**2.4.7 Modelos generativos . . . . . 33**
**2.4.8 GNNs: pré-treinamento em grafos e *fine-tuning* por tarefa . . . . . 33**
**2.4.9 Meta-aprendizado e *Few-shot* . . . . . 33**
**2.4.10 Aprendizado por reforço . . . . . 34**
**2.4.11 Padrões recorrentes . . . . . 34**
**2.4.12 Limitações, desvantagens e lacunas . . . . . 34**
2.5 Campos de aplicação e algumas tendências atuais na literatura . . . . . 35
**2.5.1 Modelos fundacionais e adaptação eficiente . . . . . 36**
**2.5.2 Auto-supervisão e transferibilidade . . . . . 36**
**2.5.3 *Domain adaptation* e robustez sob mudança de distribuição . . . . . 36**
**2.5.4 Saúde e imagem médica . . . . . 36**
**2.5.5 Séries temporais e aplicações industriais . . . . . 36**
**2.5.6 AT e aprendizagem auto-supervisionada . . . . . 37**
2.6 Posicionamento do AT com FAM quanto a limitações e popularidade . . . . 37

**3 ARTMAP FUZZY . . . . . 38**
3.1 FUNDAMENTOS DA TEORIA DA RESSONÂNCIA ADAPTATIVA (ART) 38
3.2 FUZZY ART: NOTAÇÃO E MECANISMO DE CATEGORIZAÇÃO . . . . 39

**3.2.1 Codificação complementar** . . . 39
**3.2.2 Operador *fuzzy AND* e norma $L_1$** . . . 39
**3.2.3 Pesos (protótipos) e inicialização** . . . 39
**3.2.4 Função de escolha** . . . 40
**3.2.5 Teste de vigilância (*match*) e busca** . . . 40
**3.2.6 Regra de aprendizado** . . . 40
3.3 ARQUITETURA da rede ARTMAP Fuzzy . . . 41
3.4 APRENDIZADO SUPERVISIONADO . . . 41
**3.4.1 *Match tracking*** . . . 41
**3.4.2 Atualização do campo de mapeamento** . . . 42
**3.4.3 Procedimento de treinamento** . . . 42
**3.4.4 Predição** . . . 42
3.5 Análise interpretativa do uso das matrizes de pesos na transferência entre domínios . . . 42
**3.5.1 O que representam $\mathbf{W}^a$ e $\mathbf{W}^{ab}$?** . . . 43
**3.5.2 O que significa transferir $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ entre domínios?** . . . 43
**3.5.3 Resultados esperados sob diferentes cenários de transferência** . . . 44
**3.5.4 Discussão: por que a transferência pode funcionar (ou falhar) na FAM** . . . 44
3.6 USO DE FAM EM AT E DOMAIN ADAPTATION NA LITERATURA . . . 44
**3.6.1 Transferência por reutilização de representação** . . . 45
**3.6.2 Aprendizado incremental e adaptação temporal** . . . 46
**3.6.3 Extensões para robustez sob mudança de domínio** . . . 46
**3.6.4 Aplicações com condições variáveis e cenários abertos** . . . 46
**3.6.5 Síntese** . . . 46

**4 MATERIAIS E MÉTODOS** . . . 47
4.1 Bases de dados utilizadas . . . 47
**4.1.1 Occupancy Detection** . . . 49
**4.1.2 Room Occupancy Estimation** . . . 49
**4.1.3 Análise do uso das bases em aprendizado por transferência** . . . 50
4.2 Definição dos domínios . . . 52
**4.2.1 Domínio-fonte $\mathbf{S}$** . . . 52
**4.2.2 Domínio-alvo $\mathbf{T}_1$** . . . 52
**4.2.3 Domínio-alvo sintético $\mathbf{T}_2(\lambda)$** . . . 52
4.3 Harmonização de atributos . . . 53
4.4 Pré-processamento . . . 53
**4.4.1 Normalização e consistência** . . . 53
**4.4.2 Particionamento treino/teste e amostragem por fração** . . . 54
4.5 Geração de *shift* e construção de $\mathbf{T}_2(\lambda)$ . . . 55

4.6 Treinamento no domínio-fonte $\mathbf{S}$ . . . 56
4.7 Medição de distâncias entre domínios . . . 56
**4.7.1 Proxy A-Distance (PAD)** . . . **56**
**4.7.2 Maximum Mean Discrepancy (MMD)** . . . **56**
**4.7.3 CORAL (Correlation Alignment)** . . . **57**
4.8 Estratégias de transferência de aprendizado . . . 57
**4.8.1 Baselines** . . . **57**
4.8.1.1 ZS (*Zero-Shot*) . . . 58
4.8.1.2 TFS (*Train From Scratch*) . . . 58
**4.8.2 Transferência direta com *fine-tuning* controlado** . . . **58**
4.8.2.1 $\mathrm{AT}_{\mathrm{Wa}}$(adaptação restrita) . . . 58
4.8.2.2 $\mathrm{AT}_{\mathrm{Wab}}$(adaptação ampliada) . . . 59
**4.8.3 Transferência Seletiva (Top-K) (Selective Transfer Tuning (Top-K))** . . . **59**
**4.8.4 Poda por pureza** . . . **60**
4.9 Análise de sensibilidade aos hiperparâmetros . . . 61
4.10 Protocolo experimental e curvas de aprendizado . . . 61
4.11 Métricas de avaliação . . . 61
4.12 Considerações de implementação e reprodutibilidade . . . 62

**5 RESULTADOS E DISCUSSÃO . . . 63**
5.1 Distâncias entre domínios e efeito de $\lambda$ . . . 63
5.2 Resultados no domínio alvo ($\mathbf{T}_1$) . . . 64
**5.2.1 Principais resultados em $\mathbf{T}_1$** . . . **64**
**5.2.2 Efeito da fração de dados rotulados empregada** . . . **67**
5.3 Resultados no domínio alvo ($\mathbf{T}_2(\lambda)$) . . . 67
**5.3.1 Robustez da acurácia da FAM** . . . **69**
**5.3.2 Complexidade vs. desempenho ao variar $\lambda$** . . . **72**
**5.3.3 Curvas de aprendizado no domínio $\mathbf{T}_2(\lambda)$** . . . **72**
5.4 Transferência seletiva (Top-k e poda) . . . 73
5.5 Sensibilidade a $\rho$ e $\beta$ . . . 73
5.6 Síntese dos resultados . . . 75

**6 CONCLUSÃO . . . 77**
6.1 Principais conclusões . . . 77
6.2 Contribuições do trabalho . . . 78
6.3 Limitações . . . 79
6.4 Trabalhos futuros . . . 79

**REFERÊNCIAS . . . 80**

**DADOS CURRICULARES . . . . . 87**

# 1 INTRODUÇÃO

Muitos métodos de aprendizado de máquina (AM) funcionam bem apenas sob a suposição comum de que os dados de treinamento e teste possuem a mesma distribuição estatística. Quando a distribuição muda, muitos modelos são treinados novamente, do zero, utilizando novos dados de treinamento, para que seja possível obter bons resultados. Em muitas aplicações, treinar o modelo do zero pode ser muito difícil, pelos altos custos ou, até mesmo, impossível, pela necessidade de coletar e rotular novos dados para o treinamento. Nesse sentido, é importante reduzir a necessidade e o esforço de coletar novos dados de treinamento utilizando conhecimento previamente adquirido em uma tarefa de aprendizado como ponto de partida para outra tarefa relacionada. A esse tipo de estratégia dá-se o nome de transferência de conhecimento ou aprendizado por transferência.

O aprendizado por transferência tem como objetivo resolver a tarefa de destino aproveitando o conhecimento adquirido em uma ou mais tarefas de origem em diferentes domínios, de modo que não é necessário aprender do zero com uma quantidade muito grande de dados. Dessa forma, o aprendizado por transferência pode lidar com o problema de insuficiência de dados de treinamento rotulados e diminuir o tempo e os recursos necessários para treinar um modelo. Também pode ajudar a resolver a possível degradação no desempenho de modelos de AM causada pela incompatibilidade de distribuição, fundindo o conhecimento de um ou vários domínios diferentes (Pan; Yang, 2010; Perkins; Salomon, 1992; Zhuang *et al.*, 2021).

A origem do conceito do aprendizado por transferência pode ser atribuída a psicologia educacional. De acordo com a teoria da generalização da transferência, proposta pelo psicólogo C. H. Judd em 1908, a aprendizagem por transferência é o resultado da generalização da experiência. É possível realizar a transferência de uma situação para outra, desde que a pessoa generalize sua experiência. De acordo com essa teoria, o pré-requisito da transferência é a necessidade de haver uma conexão entre duas atividades de aprendizagem. Pode-se citar, como exemplos intuitivos da transferência de conhecimentos, a aprendizagem de um instrumento de corda para outro, aprender a andar de motocicleta quando já se sabe andar de bicicleta, aprender a língua espanhola quando já se sabe a língua portuguesa, entre outros (Judd, 1908; Zhuang *et al.*, 2021).

O conhecimento transferido nem sempre tem um impacto positivo nas novas tarefas. Se houver pouco em comum entre os domínios, a transferência de conhecimento pode não ser bem-sucedida. Além disso, as semelhanças entre os domínios nem sempre facilitam o aprendizado, porque às vezes as semelhanças podem induzir erros. Por exemplo, embora o espanhol e o português tenham uma relação estreita entre si e ambos pertençam ao grupo de idiomas românicos, as pessoas que falam espanhol podem ter dificuldades específicas no aprendizado do português advindas do conhecimento do espanhol, como as palavras falsas cognatas.

No campo da psicologia, o fenômeno no qual a experiência anterior tem um efeito negativo

no aprendizado de novas tarefas é chamado de transferência negativa. Da mesma forma, na área de aprendizado de máquina, se o domínio de destino for afetado negativamente pelo conhecimento transferido, o fenômeno também é chamado de transferência negativa. A ocorrência de transferência negativa pode depender de vários fatores, como a relevância entre os domínios de origem e de destino e a capacidade de aprendizado do modelo utilizado em encontrar a parte transferível e benéfica do conhecimento entre os domínios (Pan; Yang, 2010; Perkins; Salomon, 1992; Wang *et al.*, 2019).

Pesquisas em aprendizado por transferência e áreas semelhantes parecem ter tido início em 1992, porém, com uma variedade de nomenclaturas e abordagens, como *learning to learn, life-long learning, knowledge transfer, inductive transfer, multitask learning, knowledge consolidation, context-sensitive learning, knowledge-based inductive bias, metalearning, and incremental/cumulative learning* (Thrun; Pratt, 1998). Um trabalho interessante na área foi o de Pan e Yang (2010), que ajudou a agrupar e categorizar as diferentes abordagens e áreas semelhantes, além de consolidar nomenclaturas e definições na área de pesquisa desde então.

Os principais modelos ou tipos de técnicas comumente utilizados na literatura de aprendizado por transferência são modelos "rasos"(com poucas camadas, menos frequentes recentemente), que incluem redes neurais aritificias "rasas"e outros algoritmos, CNN/Visão computacional, Transformers, PEFT (*parameter-efficient fine-tuning*), adaptação de domínio, modelos generativos, GNN (grafos), *few-shot*/meta-aprendizado e aprendizado por reforço.

O poder de generalização dos modelos de aprendizado utilizados para extrair o conhecimento do domínio de origem é muito importante para o sucesso da transferência. A rede ARTMAP *Fuzzy* foi utilizada para teste do potencial da transferência estrutural de conhecimento, utilizando a própria estrutura interna de armanezamento de conhecimento, com variações de estratégias de transferências diretas e seletivas, de um domínio de origem (ou fonte) para dois domínios de destino (ou alvo), um real e um sintético. O domínio fonte é um *dataset* de contagem de ocupação a partir de dados de sensores, o alvo real é uma tarefa de classificação binária da ocupação, com domínio diferente do de origem, e o sintético foi gerado com diferentes deslocamentos a partir do alvo real.

O objetivo principal do trabalho é propor diferentes metodologias para a realização do aprendizado por transferência na rede neural ARTMAP *Fuzzy*, utilizando sua capacidade de aprendizado incremental e robustez, para transferir o aprendizado entre os domínios alvo real e o sintético, com diferentes graus de deslocamento, criando condições estruturais adequadas para que a transferência seja bem-sucedida, realizando um tipo de transferência estrutural, aparentemente, não realizada na literatura disponível, para FAM, com diferentes estrátégias de transferência. São realizadas também, análises do efeito da quantidade de dados rotulados do domínio fonte empregados na transferência, do efeito do deslocamento e análise de sensibilidade em relação aos parâmetros de vigilância e de aprendizado da rede ARTMAP *Fuzzy*.

# 2 APRENDIZADO POR TRANSFERÊNCIA

Em aplicações reais de AM é recorrente que o conjunto de dados disponível para o problema de interesse (domínio-alvo) seja limitado, caro de rotular ou sofra variações estatísticas relevantes em relação aos dados usados no desenvolvimento do modelo (por exemplo, mudança de instrumento/sensor, população, ambiente ou tempo). Nessas condições, modelos treinados sob a hipótese de estacionariedade podem degradar seu desempenho quando aplicados em novos cenários, fenômeno frequentemente descrito como *dataset shift* (Quiñonero-Candela *et al.*, 2009).

A família de técnicas conhecida como *transfer learning*, ou, em português, aprendizado por transferência (AT), lida com esse desafio explorando o conhecimento previamente adquirido em um domínio e/ou tarefa de origem (ou fonte) para melhorar o aprendizado no domínio de destino (ou alvo). Além de reduzir a necessidade de dados rotulados no alvo, o AT pode acelerar o treinamento (por meio de estratégias de inicialização), aumentar a robustez sob *shift* e viabilizar problemas para os quais não há dados suficientes para treinamento do zero (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016).

Historicamente, a reutilização de conhecimento dialoga com a discussão clássica sobre *viés indutivo* e a necessidade de suposições adicionais para generalização além dos dados observados. Na literatura especializada recente, o AT consolidou-se como uma área transversal e multidisciplinar, que inclui *domain adaptation*, *multi-task learning*, pré-treinamento e ajuste fino em redes profundas e, mais recentemente, adaptação de modelos fundacionais (Bommasani *et al.*, 2021; Mitchell, 1980; Pan; Yang, 2010).

## 2.1 DIFERENÇAS ENTRE O AT E O AM TRADICIONAL

No aprendizado de máquina (AM) tradicional, assume-se tipicamente que os exemplos usados no treinamento e aqueles observados na fase de teste são amostras independentes e identicamente distribuídas (*i.i.d.*) e, portanto, pertencem ao mesmo domínio e à mesma tarefa, *i.e.*, são gerados pela mesma distribuição estatística. No caso supervisionado, essa hipótese é expressa pela igualdade entre as distribuições conjuntas de treino e teste:

$$P_{\text{treino}}(X, Y) = P_{\text{teste}}(X, Y). \tag{1}$$

Nessa expressão, $X$ denota a variável aleatória de entrada (atributos, *features*) e $Y$ denota a variável aleatória de saída (rótulo). O termo $P_{\text{treino}}(X, Y)$ representa a distribuição conjunta que gera pares $(X, Y)$ durante o treinamento, enquanto $P_{\text{teste}}(X, Y)$ representa a distribuição conjunta que gera os pares na etapa de teste. Assim, (1) mostra que, em ambas as fases, observações e rótulos são produzidos pelo mesmo mecanismo probabilístico.

Sob a hipótese da equação (1), estima-se um preditor a partir do conjunto de treinamento por meio da minimização do risco empírico (ERM), do inglês *empirical risk minimization* (Pan; Yang, 2010). Considerando uma classe de hipóteses $\mathcal{F}$ (o conjunto de modelos candidatos) e uma função de perda $\mathcal{L}(\cdot, \cdot)$ (que quantifica o erro de predição), define-se:

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\big(f(x_i), y_i\big). \tag{2}$$

Na Equação (2) $\mathbf{n}$ é o número de amostras de treinamento e $\{(x_i, y_i)\}_{i=1}^{n}$ é o conjunto de dados observado, no qual $\mathbf{x}_i$ é a $\mathbf{i}$-ésima entrada e $\mathbf{y}_i$ é o rótulo correspondente. A função $f(\cdot)$ é um modelo candidato pertencente a $\mathcal{F}$ e $f(x_i)$ é a predição do modelo para a amostra $\mathbf{x}_i$. A perda $\mathcal{L}\big(f(x_i), y_i\big)$ mede o custo associado a prever $f(x_i)$ quando o valor verdadeiro é $\mathbf{y}_i$ (por exemplo, erro de classificação ou erro quadrático). O termo $\frac{1}{n}\sum_{i=1}^{n} \mathcal{L}\big(f(x_i), y_i\big)$ corresponde ao risco empírico, isto é, à média das perdas no conjunto de treinamento. Por fim, o operador $\arg\min_{f \in \mathcal{F}}$ seleciona a hipótese $\hat{f}$ que minimiza esse risco empírico dentre todos os modelos em $\mathcal{F}$.

No AT, a Equação (1) é explicitamente relaxada: admite-se que o domínio e/ou a tarefa no alvo diferem da fonte. Em termos práticos, o AT pode alterar a forma como dados da fonte e do alvo entram no treinamento, como os parâmetros são inicializados/regularizados e/ou como a representação é construída, com o objetivo de reduzir o custo amostral e mitigar degradações decorrentes de *shift* (Pan; Yang, 2010; Quiñonero-Candela *et al.*, 2009; Weiss; Khoshgoftaar; Wang, 2016).

De forma consistente na literatura, o AT é particularmente indicado quando há poucos dados rotulados no alvo e existe um domínio-fonte com dados abundantes (rotulados ou não) e relação semântica/estatística relevante com o alvo, também em aplicações nas quais rotular dados no alvo é caro, demorado ou inviável (por restrições operacionais, éticas ou regulatórias), ou, por fim, quando existe *domain shift* e é necessário adaptar um modelo a uma nova população, instrumento, ambiente ou período temporal (Pan; Yang, 2010; Quiñonero-Candela *et al.*, 2009; Sugiyama; Kawanabe, 2012; Weiss; Khoshgoftaar; Wang, 2016).

Por outro lado, o AT pode não ser recomendado quando a relação entre fonte e alvo é fraca ou inexistente, elevando o risco de *negative transfer*, ou transferência negativa, (transferência que degrada o desempenho no alvo) (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016), também quando há dados rotulados abundantes e representativos no alvo, tornando o ganho marginal do AT pequeno frente ao treinamento direto no alvo ou quando o custo de engenharia/validação (seleção de fonte, ajuste de hiperparâmetros, avaliação de robustez) supera os benefícios esperados.

Resultados teóricos podem fornecem intuição sobre quando a transferência pode ser bem-sucedida. Um limite clássico estabelece, para um classificador $h$ em uma classe $\mathcal{H}$, um majorante para o erro no alvo em função do erro na fonte e de uma medida de divergência entre domínios

(Ben-David *et al.*, 2010):

$$\epsilon_T(h) \leq \epsilon_S(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(S, T) + \lambda, \tag{3}$$

em que $\epsilon_S$ e $\epsilon_T$ denotam erros (riscos) em fonte e alvo, $d_{\mathcal{H}\Delta\mathcal{H}}(S, T)$ é a divergência induzida por $\mathcal{H}\Delta\mathcal{H}$ entre as distribuições de fonte e alvo, e $\lambda$ captura o erro do melhor classificador compartilhado entre os domínios. A inequação (3) evidencia dois pontos práticos: reduzir divergência entre domínios é crucial, e se as tarefas/domínios forem intrinsecamente incompatíveis (alto $\lambda$), a transferência pode falhar ou ser contraproducente (transferência negativa) (Ben-David *et al.*, 2010; Pan; Yang, 2010).

Entre as principais vantagens do AT, de acordo com a literatura, destacam-se maior eficiência amostral, aceleração do treinamento, melhoria de desempenho em regimes de poucos dados e maior robustez sob *shift*. Entre as principais desvantagens, destacam-se o risco de transferência negativa, necessidade de avaliar compatibilidade entre domínios/tarefas e possibilidade de introdução de vieses do domínio-fonte no alvo. Em modelos de grande escala, também surgem desafios adicionais ligados a custo computacional, reprodutibilidade e governança do processo de adaptação (Bommasani *et al.*, 2021; Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016).

## 2.2 BREVE HISTÓRICO E PRIMEIROS TRABALHOS

Embora o termo "*transfer learning*" tenha se consolidado posteriormente, a ideia de reutilização de conhecimento está ligada a debates clássicos sobre viés e generalização e, no contexto de redes neurais, um marco inicial frequentemente citado é o trabalho de Pratt, que investiga a reutilização de parâmetros treinados em uma tarefa correlata para acelerar o aprendizado em uma tarefa-alvo (Mitchell, 1980; Pratt, 1992).

Os modelos referidos por *lifelong learning* e *learning to learn* também contribuíram de maneira relevante, em particular, ao tratar o aprendizado como um processo ao longo de múltiplas tarefas e ambientes. Um exemplo representativo pode ser encontrado no trabalho de Thrun e Mitchell, de 1995, (Thrun; Mitchell, 1995), que discute o aprendizado robótico como um problema permanente (ao longo da vida) e enfatiza a reutilização de conhecimento adquirido em tarefas anteriores.

Outro marco central é o *multi-task learning* (MTL), que explicita o compartilhamento de representações e indutivamente transfere informações entre tarefas relacionadas e, do ponto de vista teórico, fornece base conceitual para meta-aprendizado e transferência por meio de um modelo formal para aprendizado de viés em ambientes de tarefas relacionadas (Baxter, 2000; Caruana, 1997).

Com a consolidação do campo, surveys como (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016; Zhuang *et al.*, 2021) sistematizaram definições, cenários, estratégias e riscos (incluindo transferência negativa) e conectaram AT a problemas de *domain adaptation* e *dataset shift*.

## 2.3 CONCEITOS FUNDAMENTAIS E NOTAÇÃO CLÁSSICA

No contexto do aprendizado por transferência, um *domínio* $\mathcal{D}$ é composto por dois componentes: um espaço de atributos $\mathcal{X}$ e uma distribuição de probabilidade marginal $P(X)$, em que o vetor de atributos $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$, denota o conjunto de atributos do domínio (Pan; Yang, 2010; Zhuang *et al.*, 2021), sendo $\mathbf{n}$ o número total de atributos. Assim, denota-se o domínio por $\mathcal{D} = \{\mathcal{X}, P(X)\}$. Dessa forma, se dois domínios forem diferentes, eles podem apresentar espaços de atributos distintos ($\mathcal{X}$) e/ou distribuições marginais diferentes $P(X)$.

Dado um domínio $\mathcal{D}$, uma *tarefa* $\mathcal{T}$ é composta por um espaço de rótulos $\mathcal{Y}$ e uma função preditiva $f(.)$, que não é observável, mas pode ser retirada do par $\{x_i, y_i\}$, com $x_i \in X$ e $y_i \in \mathcal{Y}$. A função $f(.)$ é usada para calcular o rótulo $f(x)$ correspondente a uma nova instância $\mathbf{x}$. De um ponto de vista probabilístico, $f(x)$ pode ser escrita como $P(y|x)$. Assim, denota-se a tarefa por $\mathcal{T} = \{\mathcal{Y}, f(.)\} = \{\mathcal{Y}, P(y|x)\}$ (Pan; Yang, 2010).

Para apresentar algumas definições necessárias, considerar-se-á, por simplicidade e por ser mais popular na literatura, o caso em que há um domínio de origem (*source domain*) $\mathcal{D}_\mathcal{S}$ e um domínio de destino (*target domain*) $\mathcal{D}_\mathcal{T}$.

Os dados do domínio de origem são denotados por $D_S = \{(x_{S_1}, y_{S_1}), \ldots, (x_{S_{n_S}}, y_{Sn_S})\}$, em que $x_{S_i} \in \mathcal{X}_\mathcal{S}$ corresponde a instância de entrada e $y_{S_i} \in \mathcal{Y}_\mathcal{S}$ o rótulo correspondente. Similarmente, denotam-se os dados do domínio de destino por $D_T = \{(x_{T_1}, y_{T_1}), \ldots, (x_{T_{n_S}}, y_{Tn_T})\}$, em que a entrada $x_{T_i} \in \mathcal{X}_\mathcal{T}$ e $y_{T_i} \in \mathcal{Y}_\mathcal{T}$ é a saída correspondente. Em geral, $0 \leq n_T \ll n_S$ (Pan; Yang, 2010).

**Definição 1 (Aprendizado por transferência)** *Dado um domínio de origem $\mathcal{D}_\mathcal{S}$ e uma tarefa de origem $\mathcal{T}_\mathcal{S}$, um domínio de destino $\mathcal{D}_\mathcal{T}$ e uma tarefa de destino $\mathcal{T}_\mathcal{T}$, o aprendizado por transferência objetiva melhorar a aprendizagem da função preditiva alvo $f_T(.)$ no $\mathcal{D}_\mathcal{T}$ usando o conhecimento do $\mathcal{D}_\mathcal{S}$ e da $\mathcal{T}_\mathcal{S}$, desde que $\mathcal{D}_\mathcal{S} \neq \mathcal{D}_\mathcal{T}$ ou $\mathcal{T}_\mathcal{S} \neq \mathcal{T}_\mathcal{T}$ (Pan; Yang, 2010; Zhuang et al., 2021).*

Como o domínio é o par $\mathcal{D} = \{\mathcal{X}, P(X)\}$, a condição $\mathcal{D}_\mathcal{S} \neq \mathcal{D}_\mathcal{T}$ implica que, para que os domínios de origem e destino sejam diferentes , $\mathcal{X}_\mathcal{S} \neq \mathcal{X}_\mathcal{T}$ ou $P_S(X) \neq P_T(X)$. Similarmente, como uma tarefa é definida pelo par $\mathcal{T}_\mathcal{T} = \{\mathcal{Y}, P(y|x)\}$, a condição $\mathcal{T}_\mathcal{S} \neq \mathcal{T}_\mathcal{T}$ implica que, para que as tarefas de origem e destino sejam diferentes, $\mathcal{Y}_\mathcal{S} \neq \mathcal{Y}_\mathcal{T}$ ou $P_S(Y_S|X_S) \neq P_T(Y_T|X_T)$. Quando $\mathcal{D}_\mathcal{S} = \mathcal{D}_\mathcal{T}$ e $\mathcal{T}_\mathcal{S} = \mathcal{T}_\mathcal{T}$, trata-se do aprendizado de máquina tradicional (Pan; Yang, 2010; Zhuang *et al.*, 2021).

É importante ressaltar, de acordo com a Definição 1, que o aprendizado por transferência tem foco em melhorar a aprendizagem da função preditiva no $\mathcal{D}_\mathcal{T}$ e não no $\mathcal{D}_\mathcal{S}$. Deve ser elaborada uma estratégia para transferir o conhecimento de um ou mais domínios de origem para o de destino. Para que essa transferência seja possível e bem-sucedida, os domínios ou tarefas devem possuir alguma relação entre si, e os modelos utilizados devem possuir boa capacidade de generalização (Zhuang *et al.*, 2021).

Na literatura, o caso em que $\mathcal{X}_\mathcal{S} \neq \mathcal{X}_\mathcal{T}$ é comumente denominado aprendizado por transferência (AT) heterogêneo, e o caso em que $\mathcal{X}_\mathcal{S} = \mathcal{X}_\mathcal{T}$, AT homogêneo. Existem algumas estratégias e implementações comuns a todos os casos. A maioria das soluções de AT homogênea emprega três estratégias gerais que incluem tentar corrigir a diferença de distribuição marginal no $\mathcal{D}_\mathcal{S}$, tentar corrigir a diferença de distribuição condicional no $\mathcal{D}_\mathcal{S}$ ou tentar corrigir ambas ao mesmo tempo. Já a maioria das soluções de AT heterogênea concentra-se no alinhamento dos espaços de entrada do $\mathcal{D}_\mathcal{S}$ e do $\mathcal{D}_\mathcal{T}$ com a suposição de que as distribuições de domínio são as mesmas. Se as distribuições de domínio não forem iguais, serão necessárias outras etapas de adaptação de domínio (Day; Khoshgoftaar, 2017).

Além dessa classificação generalista que leva em consideração apenas os espaços de atributos, há outras classificações nas diferentes técnicas de AT baseadas no conhecimento a ser transferido, nas diferentes combinaçoes possíveis entre $\mathcal{D}_\mathcal{S}$, $\mathcal{D}_\mathcal{T}$, $\mathcal{T}_\mathcal{S}$ e $\mathcal{T}_\mathcal{T}$ e na disponilidade de rótulos em $\mathcal{D}_\mathcal{S}$ e $\mathcal{T}_\mathcal{S}$. Essas diferenciações e diferentes são importantes, pois cada uma leva a necessidades e diferentes estratégias de soluções. A próxima subseção tratará dessas classificações.

### 2.3.1 Classificação das técnicas de aprendizado por transferência

No AT, existem três questões principais de pesquisa a serem consideradas, a saber: o que transferir, como transferir e quando transferir (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016; Zhuang *et al.*, 2021).

"O que transferir" se refere a qual parte do conhecimento pode ser transferida entre domínios ou tarefas. Alguns conhecimentos são específicos para domínios ou tarefas individuais, e outros podem ser comuns entre domínios diferentes, de modo que podem ajudar a melhorar o desempenho do domínio ou da tarefa de destino. Depois de descobrir qual conhecimento pode ser transferido, é necessário desenvolver algoritmos de aprendizado para transferir o conhecimento, o que corresponde à questão "como transferir" (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016; Zhuang *et al.*, 2021).

"Quando transferir" pergunta em que situações a transferência deve ser feita. Da mesma forma, também deve-se buscar em quais situações o conhecimento não deve ser transferido. Em algumas situações, quando $\mathcal{D}_\mathcal{S}$ e $\mathcal{D}_\mathcal{T}$ não estão relacionados entre si, a transferência pode não ser bem-sucedida. Na pior das hipóteses, ela pode até prejudicar o desempenho da aprendizagem no domínio de destino, o que se refere a transferência negativa. A maioria dos trabalhos atuais sobre aprendizagem por transferência concentra-se em "o que transferir" e "como transferir", pressupondo implicitamente que os domínios de origem e de destino estejam relacionados entre si. No entanto, como evitar a transferência negativa ainda é uma importante questão em aberto (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016; Zhuang *et al.*, 2021).

Com base na Definição 1, a Tabela 1 apresenta as configurações de aprendizado por transferência resultantes das condições da definição, que foram nomeadas como AT indutivo, AT transdutivo e AT não-supervisionado, em referência a existência de rótulos em $\mathcal{D}_\mathcal{S}$ e $\mathcal{D}_\mathcal{T}$. Alguns

autores, recentemente, incluem uma quarta configuração chamada de multimodalidade, que é a junção de duas ou mais destas configurações (Niu *et al.*, 2020; Pan; Yang, 2010).

Tabela 1 – Diferenças entre as configurações de AT em relação à existência de rótulos

| **Configuração AT** | **Rótulos no $\mathcal{D}_S$** | **Rótulos no $\mathcal{D}_T$** | **Tarefas** |
|---|---|---|---|
| Indutivo | Disponíveis | Disponíveis | Regressão, Classificação |
| | Indisponíveis | Disponíveis | Regressão, Classificação |
| Transdutivo | Disponíveis | Indisponíveis | Regressão, Classificação |
| Não supervisionado | Indisponíveis | Indisponíveis | Clusterização, |
| | | | Redução de dimensionalidade |

Fonte: Adaptado de Pan e Yang (2010).

1. Aprendizado por transferência indutivo: a tarefa de destino é diferente da tarefa de origem ($\mathcal{T}_S \neq \mathcal{T}_T$), independentemente de os domínios de origem e de destino serem os mesmos ou não. Nesse caso, alguns dados rotulados no $\mathcal{D}_T$ são necessários para induzir, encontrar uma função preditiva $f_T(.)$ para uso no $\mathcal{D}_T$. Além disso, de acordo com as diferentes situações de dados rotulados e não rotulados no $\mathcal{D}_S$, o AT indutivo pode ser dividido em dois casos:

   - Muitos dados rotulados no $\mathcal{D}_S$ estão disponíveis: nesse caso, o AT indutivo é semelhante à uma área de pesquisa chamada aprendizagem multitarefa (*multitask learning*). No entanto, o AT indutivo visa apenas a obter alto desempenho na tarefa do $\mathcal{D}_T$ transferindo conhecimento do $\mathcal{D}_S$, enquanto a aprendizagem multitarefa tenta aprender as duas tarefas simultaneamente;
   - Não há dados rotulados no $\mathcal{D}_S$ disponíveis: nesse caso, o AT indutivo é semelhante à aprendizagem autodidata (*self-taught learning*) (Raina *et al.*, 2007). No aprendizado autodidata, os espaços de rótulos entre $\mathcal{D}_S$ e $\mathcal{D}_T$ podem ser diferentes, o que implica que as informações laterais de $\mathcal{D}_S$ não podem ser usadas diretamente. Portanto, é semelhante ao AT indutivo quando dados rotulados no $\mathcal{D}_S$ não estão disponíveis;

2. Aprendizado por transferência transdutivo: as tarefas de origem e de destino são iguais ($\mathcal{T}_S = \mathcal{T}_T$), enquanto os domínios de origem e de destino são diferentes ($\mathcal{D}_S \neq \mathcal{D}_T$). Nessa situação, nenhum dado rotulado no ($\mathcal{D}_T$) está disponível, enquanto muitos dados rotulados no ($\mathcal{D}_S$) estão disponíveis. Além disso, de acordo com os espaços de atributos ($\mathcal{X}_S$) e ($\mathcal{X}_T$) o AT transdutivo pode ser dividido em dois casos:

   - Os espaços de atributos entre $\mathcal{D}_S$ e $\mathcal{D}_T$ são diferentes, ($\mathcal{X}_S \neq \mathcal{X}_T$ );
   - Os espaços de atributos são os mesmos ($\mathcal{X}_S = \mathcal{X}_T$ ), mas a distribuição de probabilidade marginal entre os dados de entrada é diferente, ($P_S(X) \neq P_T(X)$);

3. Aprendizado por transferência não supervisionado: a tarefa de destino também é diferente $\mathcal{T}_S \neq \mathcal{T}_T$. No entanto, o AT não supervisionado concentra-se na solução de tarefas de

aprendizagem não supervisionadas no domínio de destino, como agrupamento, redução de dimensionalidade e estimativa de densidade. Nesse caso, não há dados rotulados disponíveis em $\mathcal{D}_S$ e $\mathcal{D}_\mathcal{T}$ no treinamento.

A Tabela 2 apresenta uma síntese das classificações, hipótese principal e estratégias típicas de transferência em cada uma delas.

Tabela 2 – Classificações clássicas de AT: cenário, hipótese principal e estratégias típicas.

| **Cenário** | **Hipótese/condição** | **Estratégias típicas** |
|---|---|---|
| Indutivo | Tarefas diferem ($\mathcal{T}_S \neq \mathcal{T}_T$); rótulos no alvo (parciais ou completos). | MTL e compartilhamento de representações; inicialização com pesos pré-treinados; regularização em torno de parâmetros-fonte; meta-aprendizado e poucos exemplos. |
| Transdutivo (DA) | Tarefas iguais ($\mathcal{T}_S = \mathcal{T}_T$); domínios diferem ($P_S(X) \neq P_T(X)$); rótulos no alvo ausentes ou escassos. | Alinhamento de distribuições; alinhamento adversarial; reponderação de instâncias; normalização/estabilização de estatísticas. |
| Não supervisionado | Ausência de rótulos no alvo; tarefas não supervisionadas. | Transferência de subespaços/estruturas latentes; redução de dimensionalidade transferível; clusterização com alinhamento entre domínios. |

Fonte: Adaptado de Pan e Yang (2010).

As abordagens para transferir a aprendizagem nos três cenários podem ser resumidas em quatro abordagens com base em "O que transferir" (Pan; Yang, 2010).

A primeira abordagem é a aprendizagem por transferência de instâncias, ou somente transferência de instâncias, que pressupõe que determinadas partes dos dados no domínio de origem podem ser reutilizadas para a aprendizagem no domínio de destino, por meio da reponderação (Dai *et al.*, 2007; Pan; Yang, 2010), visto que os domínios e tarefas, apesar de diferentes, são sempre relacionados. A reponderação de instâncias e a amostragem de importância são duas técnicas importantes nesse contexto.

A segunda abordagem é a transferência de representação de atributos. A ideia é aprender uma representação de atributos "boa" o suficiente, encontrando os atributos latentes comuns por meio da transformação de atributos e usá-los como uma ponte para transferir conhecimento para o domínio de destino. As abordagens baseadas em atributos transformam os atributos originais em novas representações de atributos para a transferência de conhecimento, objetivando minimizar a diferença de distribuição marginal e condicional, preservar as propriedades ou as estruturas potenciais dos dados e encontrar a correspondência entre os atributos (Pan; Yang, 2010).

O terceiro caso é chamado de transferência de parâmetros, que pressupõe que as tarefas de origem e destino compartilham alguns parâmetros ou distribuições prévias dos hiperparâmetros

dos modelos. O conhecimento transferido é codificado nos parâmetros ou nos antecedentes compartilhados. Assim, ao descobrir os parâmetros ou os antecedentes compartilhados, o conhecimento pode ser transferido entre as tarefas (Day; Khoshgoftaar, 2017; Pan; Yang, 2010).

A última abordagem é chamada de transferência de conhecimento relacional. Essa abordagem não pressupõe que os dados extraídos de cada domínio sejam independentes e identicamente distribuídos (i.i.d.), como tradicionalmente se supõe, e sim tenta transferir a relação entre os dados entre domínios. Esses tipos de dados podem ser representados por várias relações, como dados em rede e dados de redes sociais. Nessa abordagem as técnicas de aprendizado relacional de base estatística são predominantes (Pan; Yang, 2010; Zhuang *et al.*, 2021).

Pelas características das configurações do AT, não é possível utilizar as quatro abordagens nas três configurações, somente no indutivo, pela presença de alguns dados rotulados no $\mathcal{D}_{\mathcal{T}}$. No AT transdutivo pode-se utilizar a transferência de instâncias e de representação de atributos. No AT não supervisionado, somente é possível utilizar a transferência de representação de atributos, visto que não há rótulos em nenhum domínio.

Também se pode fazer uma combinação entre duas ou mais estratégias de transferência, apesar de não ser muito comum na literatura, visto que se deve levar em conta as particularidades da configuração de AT utilizada, a possibilidade de utilização de uma abordagem em particular, a aderência ao problema da pesquisa, além da execução da transferência em si.

## 2.4 PRINCIPAIS MODELOS E ARQUITETURAS UTILIZADOS NA LITERATURA DE AT

Em termos conceituais, as revisões clássicas disponíveis descrevem o AT pelo que é transferido (instâncias, atributos/representações, parâmetros, relações) e pelos mecanismos de transferência, como reponderação, alinhamento de representações, *fine-tuning*, treinamento multi-tarefa e *meta-learning* (Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016; Zhuang *et al.*, 2021). Na prática, a literatura contemporânea costuma organizar o campo por famílias de modelos e arquiteturas, pois elas determinam diretamente onde o conhecimento reside (atributos, pesos, módulos, memórias, políticas de controle) e como pode ser reutilizado sem incorrer em transferência negativa (piora do desempenho no alvo).

Pode-se organizar os trabalhos/modelos publicados na literatura em oito principais grupos: modelos "rasos", CNN/Visão computacional, *Transformers*, PEFT (*parameter-efficient fine-tuning*), adaptação de domínio, modelos generativos, GNN (grafos), *few-shot*/meta-aprendizado e aprendizado por reforço. A Tabela 3 resume as famílias de modelos/arquiteturas mais comuns no AT e os padrões de uso típicos na maioria trabalhos publicados e apresenta exemplos representativos/importantes de cada grupo.

Tabela 3 – Famílias de modelos/arquiteturas frequentemente usados em AT e exemplos.

| **Família de modelos** | **Uso típico em AT** | **Exemplos** |
|---|---|---|
| Modelos "rasos" (*boosting*, lineares) | Reponderação de instâncias e adaptação supervisionada/semi-supervisionada | (Blitzer; McDonald; Pereira, 2006; Dai *et al.*, 2007; III, 2007). |
| CNN (convolutional neural networks)/Visão computacional | Pré-treino em larga escala e *fine-tuning*; uso de atributos genéricos (*"deep features"*) | (Donahue *et al.*, 2014; He *et al.*, 2016; Krizhevsky; Sutskever; Hinton, 2012; Oquab *et al.*, 2014; Yosinski *et al.*, 2014). |
| *Transformers* (NLP e visão computacional) | Pré-treinamento auto-supervisionado e adaptação ao alvo (*fine-tuning*) | (Devlin *et al.*, 2019; Dosovitskiy *et al.*, 2021; Vaswani *et al.*, 2017). |
| PEFT (*parameter-efficient fine-tuning*) | Adaptação por módulos/baixo posto para reduzir custo e sobreajuste | (Houlsby *et al.*, 2019; Hu *et al.*, 2022). |
| Auto-supervisão/contraste | Aprendizado de representações gerais para múltiplos alvos | (Chen *et al.*, 2020; He *et al.*, 2020). |
| Adaptação de domínio (profundo) | Alinhamento estatístico/geométrico de atributos ou aprendizado adversarial (invariância ao domínio) | (Ganin *et al.*, 2016; Long *et al.*, 2015; Sun; Saenko, 2016; Tzeng *et al.*, 2017). |
| Modelos generativos (VAE/GAN) | Transferência por síntese/normalização de domínio (por exemplo tradução entre domínios) e regularização de representações | (Bousmalis *et al.*, 2017; Goodfellow *et al.*, 2014; Kingma; Welling, 2014; Zhu *et al.*, 2017). |
| GNN (grafos) | Pré-treino em grafos e *fine-tuning* por tarefa | (Hu *et al.*, 2020). |
| Few-shot / meta-aprendizado | Aprender a adaptar rapidamente com poucos dados (inicializações/métricas/meta-otimização) | (Finn; Abbeel; Levine, 2017; Snell; Swersky; Zemel, 2017; Vinyals *et al.*, 2016). |
| AR (aprendizado por reforço) | Transferência de políticas/representações e mitigação de esquecimento em múltiplas tarefas | (Rusu *et al.*, 2016; Taylor; Stone, 2009). |

Fonte: Autoria própria.

### 2.4.1 Modelos "clássicos" (pré-aprendizado profundo): reponderação, expansão de atributos e correspondências

Antes da dominância numérica do *deep learning*, ou aprendizado profundo (AP) nos trabalhos publicados, o AT aparece com frequência como reponderação de instâncias (quando há *covariate shift*), construção de atributos compartilhados e adaptação/reestruturação do problema para ser mais compatível com dados do alvo.

Um exemplo canônico de transferência de instâncias é o algoritmo *TrAdaBoost*, que adapta a técnica de *boosting* para misturar dados dos domínios fonte e alvo ao reponderar exemplos durante o treinamento. Já a *feature augmentation* (aumento de atributos) do algoritmo *EasyAdapt* cria versões "gerais" e "específicas de domínio" dos atributos, permitindo que um modelo linear aprenda componentes compartilhados e componentes próprios do alvo de modo simples. Outra linha expressiva nessa segamentação constrói uma representação intermediária por correspondências entre atributos de domínios distintos, como no algoritmo *Structural Correspondence*

*Learning*. Esses trabalhos também se conectam a análises teóricas de adaptação entre domínios, que formalizam condições sob as quais o erro no alvo pode ser limitado a partir do erro na fonte e de uma medida de divergência entre domínios (Ben-David *et al.*, 2010; Blitzer; McDonald; Pereira, 2006; Dai *et al.*, 2007; III, 2007).

### 2.4.2 Redes neurais profundas para visão computacional: CNNs, atributos profundos e *fine-tuning*

Em visão computacional, o AT se consolidou no padrão de executar um pré-treinamento em grande escala (por exemplo, na base de dados ImageNet) e adaptar a uma tarefa alvo por extração de atributos junto a um classificador raso; *fine-tuning* parcial e/ou total e, ainda, camadas de saídas específicas de tarefa acopladas a um extrator de características pré-treinado. Arquiteturas como *AlexNet* e *ResNet* se tornaram extratores de características comuns nessa segmentação. (He *et al.*, 2016; Krizhevsky; Sutskever; Hinton, 2012).

Trabalhos representativos mostram que ativações internas das redes CNNs funcionam como descritores genéricos, com características de ativação convolucionais profundas, que camadas pré-treinadas podem ser reutilizadas para tarefas com poucos dados (por exemplo, no modelo VOC) e que, em geral, camadas iniciais tendem a ser mais genéricas e camadas finais mais específicas da fonte, com implicações diretas para estratégias de congelamento e readaptação (Donahue *et al.*, 2014; Oquab *et al.*, 2014; Yosinski *et al.*, 2014).

### 2.4.3 Pré-treinamento auto-supervisionado e contrastivo: representações "gerais" para múltiplos alvos

Uma evolução do pré-treino supervisionado é a auto-supervisão (auto-supervisionado), na qual o extrator de características aprende representações sem rótulos explícitos, frequentemente via objetivos contrastivos. Métodos como o *SimCLR* e o *MoCo* mostraram que representações aprendidas contrastivamente transferem bem para as tarefas testadas, reduzindo dependência de rótulos na fonte (Chen *et al.*, 2020; He *et al.*, 2020). Em uma descrição mais generalista, nessa segmentação, então do AT, é feita uma substituição do pré-treino supervisionado por pré-treino auto-supervisionado seguido de readaptação no alvo.

### 2.4.4 Transformers em NLP

No campo de NLP (*natural language processing*) ou processamento de linguagem natural, o AT foi amplificado por duas ondas principais: pré-treinamento de modelos sequenciais (por exemplo, LSTM (*long short-term memory* ou memória de longo e curto prazo) e domínio dos *Transformers*. Um exemplo representativo da metodologia pré-treino e readaptação com LSTM é o algoritmo *ULMFiT*, que sistematiza etapas de adaptação, incluindo técnicas de readaptação gradual, para classificação de texto (Howard; Ruder, 2018).

No AT por meio de *Transformers*, o pré-treinamento auto-supervisionado em larga escala e a readaptação por tarefa costumam ser o padrão de metodologia mais utilizado. Assim como o *Transformer*, o modelo *BERT* pode ser citado como exemplo importante de arquitetura e regime de treino que impulsionaram o AT em NLP (Devlin *et al.*, 2019; Vaswani *et al.*, 2017).

Mais recentemente, também é observado com frequência na literatura o uso de *PEFT* (*parameter-efficient fine-tuning*), em que apenas uma pequena fração de parâmetros é atualizada. Duas abordagens emblemáticas nesse tipo de modelo são os *adapters*, ou adaptadores, módulos inseridos entre camadas para especialização por tarefa, e a técnica aplicada em *LoRA*, que adapta matrizes via decomposições de baixo posto, reduzindo custo e memória sem atualizar todos os pesos (Houlsby *et al.*, 2019; Hu *et al.*, 2022).

### 2.4.5 Transformers em visão computacional

Além das redes CNNs, os *Transformers* também ganharam espaço no campo de aplicação de visão computacional. Em um estudo significativo para o campo, a metodologia do *Vision Transformer* (ViT) obteve resultados experimentais que apontam que um *Transformer* "puro" aplicado a *patches* (blocos/recortes da imagem) pode ser altamente competitivo quando pré-treinado em grandes bases de dados e transferido para tarefas menores. Na prática, os modelos ViTs frequentemente combinam pré-treino massivo (supervisionado ou auto-supervisionado) e adaptação via *fine-tuning* completo ou PEFT (adaptadores/LoRA), especialmente quando o domínio alvo tem poucos dados (Dosovitskiy *et al.*, 2021).

### 2.4.6 Adaptação de domínio com aprendizado profundo: alinhamento estatístico adversarial

Em cenários com *domain shift* (mudança de distribuição entre fonte e alvo), técnicas de adaptação de domínio são recorrentes. Há dois blocos de técnicas muito frequentes: alinhamento estatístico e alinhamento adversarial.

O alinhamento estatístico busca reduzir, no espaço de representações, a discrepância entre as distribuições dos dados de fonte e de alvo. Conceitualmente, a ideia é aprender características que sejam simultaneamente informativas para a tarefa na fonte e mais similares entre domínios, segundo alguma medida estatística de distância/discrepância entre distribuições (por exemplo, alinhamento de momentos ou discrepâncias baseadas em *kernel*). Como exemplos representativos, podem-se citar os modelos *Deep Adaptation Networks*, que executa alinhamento por discrepâncias do tipo MMD, e *Deep CORAL*, que executa alinhamento por correlação (Ben-David *et al.*, 2010; Gretton *et al.*, 2012; Long *et al.*, 2015; Pan; Yang, 2010; Sun; Saenko, 2016).

O alinhamento adversarial, por sua vez, formula o alinhamento entre domínios como um jogo competitivo: aprende-se uma representação na qual um classificador/discriminador de domínio tem dificuldade em distinguir se uma amostra veio da fonte ou do alvo. Em termos conceituais, ao tornar as representações indistinguíveis quanto ao domínio, promove-se invariância ao domínio,

isto é, remove-se informação específica do domínio, mantendo-se ao mesmo tempo a capacidade preditiva para a tarefa. Diferentemente do alinhamento puramente estatístico, aqui a noção de discrepância é induzida pelo próprio discriminador (e pela classe de funções considerada), conectando-se naturalmente a interpretações em termos de divergências relevantes para adaptação e a variantes que lidam melhor com cenários complexos, como distribuições multimodais, entre outros (Ben-David *et al.*, 2010; Ganin *et al.*, 2016; Long *et al.*, 2018; Tzeng *et al.*, 2017).

No modelo *DANN*, um discriminador tenta prever o domínio enquanto o extrator de características aprende a dificultar essa predição, produzindo representações aproximadamente invariantes ao domínio. Outra formulação representativa é a *ADDA*, que usa treinamento adversarial com *encoders* desacoplados para fonte e alvo (Ganin *et al.*, 2016; Tzeng *et al.*, 2017).

### 2.4.7 Modelos generativos

Modelos generativos, no contexto de aprendizado de máquina, são modelos probabilísticos cujo objetivo é aprender a distribuição dos dados $p(x)$ e/ou a distribuição conjunta $p(x, y)$, em contraste com modelos discriminativos que focam diretamente em $p(y \mid x)$. Ao capturar regularidades estatísticas e dependências nos dados, modelos generativos permitem gerar novas amostras plausíveis semelhantes às observadas, imputar ou reconstruir partes faltantes, estimar verossimilhança e apoiar detecção de anomalias e aprender representações úteis para tarefas posteriores (Goodfellow *et al.*, 2014; Kingma; Welling, 2014).

Os modelos generativos são utilizados no AT, em geral, de duas maneiras principais: como regularizadores de representação, como no modelo VAEs, e como mecanismos de "tradução" de domínio, como no modelo GANs (Goodfellow *et al.*, 2014; Kingma; Welling, 2014).

### 2.4.8 GNNs: pré-treinamento em grafos e *fine-tuning* por tarefa

Em dados relacionais (grafos) o AT é utilizado, frequentemente, como pré-treino de redes GNNs (Graph Neural Networks ou Redes Neurais de Grafos), seguido de readapatação em tarefas específicas (classificação de nós, predição de *links*, classificação de grafos). Estratégias de pré-treinamento para GNNs como objetivos auto-supervisionados em grafos e protocolos de transferência são discutidos com mais detalhes em (Hu *et al.*, 2020; Scarselli *et al.*, 2009).

### 2.4.9 Meta-aprendizado e *Few-shot*

Outra família central de modelos/técnicas/arquiteturas utilizadas em AT são as baseadas em meta-aprendizado, que busca aprender inicializações, regras de atualização ou métricas que permitam adaptação rápida com poucos exemplos. O modelo *MAML* é um exemplo de meta-aprendizado baseado em otimização, aprendendo uma inicialização que se adapta com poucas etapas de gradiente. Em contraste, abordagens baseadas em métrica/memória incluem *matching networks* e *prototypical Networks*, que realizam transferência por comparação no

espaço *embedding*, frequentemente com boa estabilidade em regimes de poucos dados (Finn; Abbeel; Levine, 2017; Snell; Swersky; Zemel, 2017; Thrun; Pratt, 1998; Vinyals *et al.*, 2016).

### 2.4.10 Aprendizado por reforço

No campo de aprendizado por reforço (AR), o AT envolve reaproveitar representações, habilidades ou políticas para acelerar aprendizado em novas tarefas. Uma revisão clássica sintetiza mecanismos e desafios, como definição de similaridade entre tarefas e transferência negativa. No AR profundo, o modelo *Progressive Neural Networks* exemplifica uma arquitetura que promove transferência via conexões laterais entre "colunas" treinadas em tarefas distintas, ao mesmo tempo mitigando esquecimento catastrófico (Rusu *et al.*, 2016; Taylor; Stone, 2009).

### 2.4.11 Padrões recorrentes

De forma transversal às famílias de métodos/arquiteturas/estratégias apresentados acima, os trabalhos disponíveis na literatura aplicada tipicamente variam ao longo dos eixos:

- O que é transferido: pesos (*fine-tuning*), atributos congelados, módulos, alinhamento de distribuições (DA), ou políticas (AR) (Pan; Yang, 2010; Zhuang *et al.*, 2021);
- Granularidade: congelar o extrator de características vs. ajustar parcialmente vs. ajustar totalmente; ou em PEFT, ajustar apenas projeções/posto(*"ranks"*)/módulos (Houlsby *et al.*, 2019; Hu *et al.*, 2022);
- Risco de transferência negativa: é maior quando fonte e alvo são muito distintos e mitigado por seleção de camadas, alinhamento explícito ou meta-aprendizado (Ben-David *et al.*, 2010; Yosinski *et al.*, 2014).

### 2.4.12 Limitações, desvantagens e lacunas

Pode-se citar como limitações, principais e recorrentes, apontadas na literatura atualizada e em avaliações empíricas, as elencadas nos itens a seguir, além de lacunas abertas de pesquisa:

- Transferência negativa e seleção do domínio fonte: a transferência pode piorar o desempenho quando há incompatibilidade entre domínios/tarefas, ou quando o mecanismo de transferência impõe vieses inadequados. A caracterização, detecção e mitigação do risco de transferência negativa é o tema exclusivo de um trabalho de revisão recente (Zhang *et al.*, 2023);
- Sensibilidade a mudança de domínio e protocolos de avaliação: muitos ganhos relatados dependem fortemente do modelo selecionado, da arquitetura e dos critérios de validação. Sob protocolos consistentes, estratégias simples como ERM podem ser muito competitivas. O estudo de benchmark disponível em (Gulrajani; Lopez-Paz, 2021) enfatiza essa questão

e mostra margens pequenas sobre ERM em condições controladas (Gulrajani; Lopez-Paz, 2021). Evidências adicionais indicam que a incorporação de teorias clássicas, como medidas de discrepância, sem adaptações significativas, podem ter poder preditivo limitado para explicar generalização fora do domínio em redes modernas (Vedantam; Lopez-Paz; Schwab, 2021);

- Esquecimento catastrófico e não-estacionariedade: quando a adaptação ocorre em uma sequência de tarefas/domínios, estudos práticos mostraram degradação em tarefas antigas sem acesso a dados anteriores. Estudo de revisões recentes apontam que ainda não há consenso definitivo sobre avaliação e solução geral deste problema (Aleixo *et al.*, 2024; Wang *et al.*, 2024a; Zhou *et al.*, 2024);

- Custos elevados e barreiras de reprodutibilidade: o pré-treinamento e *fine-tuning* (especialmente em grande escala) apresentam custos computacionais, financeiros e ambientais relevantes, afetando a equidade das aplicações e dificultando a replicação dos resultados (Strubell; Ganesh; McCallum, 2019);

- A estratégia PEFT não é uma solução total e definitiva: embora reduza custo/memória, pode haver *trade-offs* de desempenho sob grandes deslocamentos de distribuição, além de desafios práticos (por exemplo, compatibilidade/portabilidade de adaptações e composição de múltiplas adaptações). Trabalhos de revisão e estudos recentes discutem essas limitações (Wang *et al.*, 2025; Wang *et al.*, 2024b).

- Governança de dados e privacidade em modelos pré-treinados: a adaptação herda riscos do pré-treino (por exemplo, vieses e vazamento/memorização de dados), que geram questionamentos sobre o uso do AT em aplicações sensíveis (Bommasani *et al.*, 2021; Carlini *et al.*, 2021).

De forma geral, a literatura aponta a necessidade de métricas e critérios de transferibilidade mais adequados para decidir quando transferir sem incorrer em transferência negativa, de padronização de protocolos e validação sob mudança realista de domínios e de tornar o AT mais robusto em regimes contínuos/não-estacionários (Gulrajani; Lopez-Paz, 2021; Vedantam; Lopez-Paz; Schwab, 2021; Wang *et al.*, 2024a; Zhang *et al.*, 2023; Zhou *et al.*, 2024).

## 2.5 CAMPOS DE APLICAÇÃO E ALGUMAS TENDÊNCIAS ATUAIS NA LITERATURA

O espectro de aplicações de AT expandiu-se substancialmente com a popularização de pré-treinamentos auto-supervisionados e a emergência de modelos fundacionais, isto é, modelos treinados em grande escala e posteriormente adaptados a tarefas por *fine-tuning* ou outras técnicas com computação mais "leve", ou menos complexa (Bommasani *et al.*, 2021). A seguir, sintetizam-se campos com maiores volumes de experimentação recente na literatura especializada.

### 2.5.1 Modelos fundacionais e adaptação eficiente

Há um campo que discute como adaptar modelos de grande escala a tarefas específicas equilibrando custo, desempenho e robustez. Nesse contexto, pode-se citar o trabalho de Wang *et.al*, de 2025, com uma proposta que aparenta reduzir a necessidade de atualizar todos os parâmetros durante a adaptação, mantendo qualidade competitiva em diferentes cenários (Wang *et al.*, 2025).

### 2.5.2 Auto-supervisão e transferibilidade

A avaliação sistemática de quão bem representações auto-supervisionadas transferem para tarefas posteriores tem recebido atenção significativa. Cita-se aqui, um estudo comparativo que analisa em quais condições pré-treinamentos auto-supervisionados superam pré-treinamentos supervisionados, e como características do domínio-alvo afetam a transferibilidade (Ericsson *et al.*, 2021).

### 2.5.3 *Domain adaptation* e robustez sob mudança de distribuição

Em aplicações com domínios distintos e poucos ou nenhum rótulo no alvo, *domain adaptation* (adaptação de domínio) permanece um tópico central. Técnicas de alinhamento de distribuições, como MMD, *Maximum Mean Discrepancy* ou Discrepância Máxima da Média, (subseção 4.7.2), CORAL, *Correlation Alignment* ou Alinhamento de Correlação, (subseção 4.7.3), bem como abordagens adversariais como DANN, (*Domain-Adversarial Neural Network* ou Rede Neural Adversarial ao Domínio) e ADDA(*Adversarial Discriminative Domain Adaptation* ou Adaptação Adversarial Discriminativa de Domínio), são bastante utilizadas e continuam sendo refinadas em diferentes configurações de rótulos e tipos de *shift*, por exemplo, nos trabalhos de (Ganin *et al.*, 2016; Pan *et al.*, 2011; Sun; Saenko, 2016; Tzeng *et al.*, 2017).

### 2.5.4 Saúde e imagem médica

Na área médica, AT e *domain adaptation* são geralmente explorados para reduzir custo de rotulagem e lidar com variações de protocolos, equipamentos e populações. Revisões específicas mostram desafios particulares do domínio (por exemplo, heterogeneidade de *scanners* e limitações de dados) e consolidam algumas linhas metodológicas para os trabalhos da área (Guan; Liu, 2022; Yu *et al.*, 2022).

### 2.5.5 Séries temporais e aplicações industriais

Em problemas industriais e de séries temporais, o *shift* é frequentemente associado a mudanças de regime, envelhecimento de equipamento, drift de sensores e variações de lote e/ou processo. Nesses cenários, técnicas de reponderação, alinhamento de atributos e estratégias híbridas são bastante empregadas para manter desempenho sob não estacionariedade, conectando

AT à literatura de *dataset shift* e *covariate shift* (Quiñonero-Candela *et al.*, 2009; Sugiyama; Kawanabe, 2012).

### 2.5.6 AT e aprendizagem auto-supervisionada

Uma revisão recente discute definições e fronteiras conceituais entre AT e aprendizagem auto-supervisionada, particularmente no contexto de pré-treinamento e reutilização de representações, (Zhao *et al.*, 2024). De forma pragmática, a auto-supervisão pode ser vista como um mecanismo de construção de representações gerais (frequentemente em grande escala), enquanto AT contempla o ato de transferir e adaptar esse conhecimento para tarefas e domínios específicos.

## 2.6 POSICIONAMENTO DO AT COM FAM QUANTO A LIMITAÇÕES E POPULARIDADE

Tendo por base os modelos/abordagens mais comuns na literatura para executar aprendizado por transferência, apresentados na subseção anterior, até o momento não foi possível encontrar nenhum trabalho na literatura que explorasse a utilização do aprendizado por transferência com a rede ARTMAP Fuzzy, especialmente no contexto da aplicação deste estudo (classificação da ocupação em um ambiente).

A luz das abordagens/técnicas e limitações identificadas pela maioria dos trabalhos publicados, apresentados nas sessões anteriores, os resultados experimentais obtidos por este estudo indicam que, no cenário estudado, a rede ARTMAP Fuzzy se mostrou um modelo robusto e confiável para executar AT por meio de transferência de parâmetros ($\mathbf{W}^a$ e $\mathbf{W}^{ab}$ em diferentes combinações), utilizando sua própria arquitetura interna, sem modificações bruscas de funcionamento. O aprendizado se manteve estável por todas as combinações de estratégia de transferência e conjuntos de parâmetros transferidos.

Além disso, o AT por meio da rede ARTMAP Fuzzy, nesse contexto, foi possível mesmo em regimes com poucos dados de treinamento. As limitações destacadas na subseção anterior incluem elevados custos computacionais, da maioria dos modelos. Então, essa pode ser entendida como uma vantagem operacional da rede ARTMAP Fuzzy aplicada ao AT, por diminuir custos computacionais. Também não foram observados, no cenário aplicado, a ocorrência de transferência negativa global nem esquecimento catastrófico, outras duas limitações importantes encontradas pelos estudos especializados.

Por fim, a rede ARTMAP Fuzzy aplicada ao AT, no cenário aplicado, não apresentou sensibilidade significativa a mudança de domínios. Foi gerado um domínio alvo sintético $\mathbf{T}_2(\lambda)$ a partir de um domínio alvo obtido de dados reais ($\mathbf{T}_1$), em que $\lambda$ mede a intensidade de deslocamento de $\mathbf{T}_1$. O aprendizado do modelo não apresentou piora com o aumento de $\lambda$, mas se manteve estável, de maneira geral, para todos os valores de $\lambda$, mostrando que a rede ARTMAP Fuzzy não apresentou sensibilidade significativa a mudança de domínio.

# 3 ARTMAP FUZZY

A rede neural artificial (RNA) ARTMAP *Fuzzy* (FAM) foi proposta por Carpenter e Grossberg (1992) como uma arquitetura de aprendizado supervisionado incremental baseada na Teoria da Ressonância Adaptativa (*Adaptive Resonance Theory* (ART)). A motivação central e base do princípio do funcionamento das RNAs da família ART é o dilema da *estabilidade–plasticidade*: como incorporar continuamente novos padrões (plasticidade) sem degradar representações previamente aprendidas (estabilidade). Na estrutura ART, a incorporação de novas informações é mediada por um mecanismo explícito de busca e por um parâmetro de vigilância, o que permite controlar o compromisso entre generalização e especificidade (Carpenter; Grossberg, 1987).

A FAM integra dois módulos do tipo *Fuzzy ART*: um módulo para processamento de entrada e formação de categorias no espaço de atributos (usualmente denotado por $\mathrm{ART}_a$) e um módulo associado ao espaço de rótulos (usualmente denotado por $\mathrm{ART}_b$), acoplados por um campo de mapeamento supervisionado (*map field*). Um mecanismo de controle interno denominado *match tracking* ajusta seletivamente a vigilância do módulo de entrada quando ocorre erro no mapeamento supervisionado, direcionando a busca para outras categorias a fim de reduzir o erro preditivo (Carpenter; Grossberg; Rosen, 1991; Carpenter *et al.*, 1992; Carpenter; Grossberg; Reynolds, 1991).

Além de apresentar a formulação clássica (notação e equações), este capítulo inclui duas seções específicas, alinhadas ao uso de FAM em *AT/domain adaptation*: (i) uma seção dedicada aos conjuntos de pesos $\mathbf{W^a}$ e $\mathbf{W^{ab}}$ e ao significado de transferi-los entre domínios; (ii) uma seção dedicada ao uso de FAM (e variantes próximas) na literatura de transferência e adaptação.

## 3.1 FUNDAMENTOS DA TEORIA DA RESSONÂNCIA ADAPTATIVA (ART)

As redes ART estabelecem o aprendizado por meio de formação de categorias com um mecanismo explícito de busca. Para uma entrada apresentada, o sistema seleciona uma categoria candidata, verifica sua compatibilidade com a entrada e, se necessário, reinicia a busca por outra categoria. Caso nenhuma categoria existente seja compatível, uma nova categoria é selecionada. Esse ciclo constitui a base de modelos ART clássicos, com diferentes escolhas para representar entradas e definir medidas de compatibilidade e aprendizagem (Carpenter; Grossberg, 1987; Grossberg, 1980).

Do ponto de vista de aprendizagem supervisionada e adaptação a dados não estacionários, três propriedades da ART são particularmente relevantes (Carpenter; Grossberg, 1987; Grossberg, 1980):

1. Aprendizado incremental: novas observações podem ser incorporadas sequencialmente, sem reprocessar todas as amostras anteriores;

2. Estabilidade por recodificação controlada: as regras de atualização restringem a forma de recodificação, reduzindo interferência destrutiva;

3. Controle de granularidade via vigilância: um parâmetro de vigilância controla a similaridade mínima exigida para ressonância, regulando o compromisso entre generalização e especificidade.

Na FAM, a vigilância assume papel adicional: ela é ajustada dinamicamente pelo *match tracking* quando ocorre erro no mapeamento supervisionado, direcionando a busca para categorias mais específicas que reduzam o erro (Carpenter; Grossberg; Reynolds, 1991).

## 3.2 FUZZY ART: NOTAÇÃO E MECANISMO DE CATEGORIZAÇÃO

A rede FAM utiliza o *Fuzzy ART* como módulo de categorização para padrões analógicos em $[0,1]^M$ (Carpenter; Grossberg; Rosen, 1991). Nesta seção, adotam-se a notação e os operadores de acordo com a formulação original.

### 3.2.1 Codificação complementar

Dada uma entrada $\mathbf{a} \in [0,1]^M$, define-se o complemento $\mathbf{a}^c$ e o vetor codificado complementarmente $\mathbf{A}$:

$$\mathbf{a}^c = \mathbf{1} - \mathbf{a}, \qquad \mathbf{A} = (\mathbf{a}, \mathbf{a}^c) \in [0,1]^{2M}, \tag{4}$$

em que $\mathbf{1}$ é um vetor unitário. Com essa codificação, obtém-se $|\mathbf{A}| = M$, isto é, a norma $L_1$ da entrada codificada é constante, o que atua como normalização e reduz proliferação de categorias (Carpenter; Grossberg; Rosen, 1991).

### 3.2.2 Operador *fuzzy AND* e norma $L_1$

O operador *fuzzy AND* (interseção *fuzzy*) é definido componente a componente:

$$(\mathbf{x} \wedge \mathbf{y})_i = \min(x_i, y_i). \tag{5}$$

A notação $|\mathbf{x}|$ denota a norma $L_1$:

$$|\mathbf{x}| = \sum_i |x_i|. \tag{6}$$

### 3.2.3 Pesos (protótipos) e inicialização

Cada categoria **j** possui um vetor de pesos (protótipos) $\mathbf{w}_j^a \in [0,1]^{2M}$. Para uma categoria não selecionada, a inicialização padrão é:

$$\mathbf{w}_j^a = \mathbf{1}. \tag{7}$$

Essa escolha torna a categoria inicialmente permissiva, indicando que não foi selecionada e permitindo capturar novos padrões e, posteriormente, especializá-los por interseção *fuzzy* (Carpenter; Grossberg; Rosen, 1991).

### 3.2.4 Função de escolha

Para cada categoria **j**, define-se:

$$T_j(\mathbf{A}) = \frac{|\mathbf{A} \wedge \mathbf{w}_j^a|}{\alpha + |\mathbf{w}_j^a|}, \qquad \alpha > 0, \tag{8}$$

em que $\alpha$ é o parâmetro de escolha. A categoria vencedora é:

$$J = \arg\max_j T_j(\mathbf{A}). \tag{9}$$

O parâmetro $\alpha$ atua na seleção de categorias e controla a sequência de busca entre os nós na camada de reconhecimento (Weenink, 1997).

### 3.2.5 Teste de vigilância (*match*) e busca

A categoria vencedora deve satisfazer o teste de vigilância:

$$\frac{|\mathbf{A} \wedge \mathbf{w}_J^a|}{|\mathbf{A}|} \geq \rho_a, \qquad \rho_a \in [0, 1]. \tag{10}$$

Se a desigualdade falha, **J** é resetada e a busca prossegue para a próxima categoria com maior ativação. Caso nenhuma categoria selecionada seja aceita, uma categoria não selecionada é selecionada e passa a representar o padrão (Carpenter; Grossberg; Rosen, 1991).

### 3.2.6 Regra de aprendizado

Após a ressonância, os pesos são atualizados por:

$$\mathbf{w}_J^{a,new} = \beta \left( \mathbf{A} \wedge \mathbf{w}_J^{a,old} \right) + (1 - \beta)\mathbf{w}_J^{a,old}, \qquad \beta \in (0, 1]. \tag{11}$$

No regime de aprendizado rápido (*fast learning*), $\beta = 1$, tem-se:

$$\mathbf{w}_J^{a,new} = \mathbf{A} \wedge \mathbf{w}_J^{a,old}. \tag{12}$$

Com $\beta = 1$, a atualização é monotônica e decrescente, favorecendo a estabilidade: os protótipos tendem a se especializar (reduzir tolerâncias por interseção), sem expandir indiscriminadamente (Carpenter; Grossberg; Rosen, 1991).

## 3.3 ARQUITETURA DA REDE ARTMAP FUZZY

A rede FAM acopla dois módulos ART e um módulo de mapeamento supervisionado (Carpenter *et al.*, 1992; Carpenter; Grossberg; Reynolds, 1991). O módulo $\mathrm{ART}_a$ recebe a entrada $\mathbf{a}$ (após normalização e codificação de complemento), forma categorias e produz um índice de categoria $\mathbf{J}$. O módulo $\mathrm{ART}_b$ representa o espaço de rótulos, isto é, codifica a resposta desejada durante treinamento. O campo de mapeamento associa cada categoria de $\mathrm{ART}_a$ a uma estrutura no espaço de rótulos, permitindo predição e aprendizado supervisionado incremental.

Nesta formulação, utiliza-se:

- $\mathbf{A} \in [0,1]^{2M}$: entrada codificada complementarmente (Eq. 4);
- $\mathbf{b} \in [0,1]^{K}$: vetor de rótulo/saída (tipicamente *one-hot encoding*);
- $\mathbf{w}_j^a \in [0,1]^{2M}$: protótipo da categoria **j** em $\mathrm{ART}_a$;
- $\mathbf{w}_j^{ab} \in [0,1]^{K}$: vetor de pesos do campo de mapeamento para a categoria **j**.

Para este trabalho, é importante ressaltar a distinção conceitual: $\mathbf{w}_j^a$ codifica uma estrutura no espaço de entrada (uma categoria/protótipo) enquanto $\mathbf{w}_j^{ab}$ codifica uma associação supervisionada entre essa estrutura e o espaço de classes (rótulos).

## 3.4 APRENDIZADO SUPERVISIONADO

Além do teste de vigilância em $\mathrm{ART}_a$ (Eq. 10), a categoria $\mathbf{J}$ também deve ser consistente com o rótulo desejado $\mathbf{b}$ e ser aceita no teste de vigilância do campo de mapeamento $\rho_{ab}$ (Carpenter *et al.*, 1992):

$$\frac{|\mathbf{b} \wedge \mathbf{w}_J^{ab}|}{|\mathbf{b}|} \geq \rho_{ab}, \qquad \rho_{ab} \in [0,1]. \tag{13}$$

### 3.4.1 *Match tracking*

Quando a categoria $\mathbf{J}$ satisfaz a vigilância em $\mathrm{ART}_a$ mas falha em produzir o rótulo correto (Eq. 13), a rede FAM aciona o mecanismo de *match tracking* (Carpenter; Grossberg; Reynolds, 1991). Em termos operacionais, eleva-se a vigilância $\rho_a$ para um valor imediatamente acima do *match* observado, forçando a busca por uma categoria mais específica:

$$\rho_a \leftarrow \frac{|\mathbf{A} \wedge \mathbf{w}_J^a|}{|\mathbf{A}|} + \epsilon, \qquad \epsilon > 0. \tag{14}$$

A categoria $\mathbf{J}$ é então *resetada* e a busca prossegue até que uma categoria mais adequada seja encontrada ou uma nova categoria seja selecionada.

### 3.4.2 Atualização do campo de mapeamento

Após selecionar uma categoria **J** consistente, os pesos do campo de mapeamento são atualizados por regra análoga à Eq. 11:

$$\mathbf{w}_J^{ab,new} = \beta_{ab}\left(\mathbf{b} \wedge \mathbf{w}_J^{ab,old}\right) + (1-\beta_{ab})\mathbf{w}_J^{ab,old}, \qquad \beta_{ab} \in (0,1]. \tag{15}$$

No regime $\beta_{ab} = 1$:

$$\mathbf{w}_J^{ab,new} = \mathbf{b} \wedge \mathbf{w}_J^{ab,old}. \tag{16}$$

### 3.4.3 Procedimento de treinamento

O procedimento de treinamento incremental para o algoritmo da FAM pode ser descrito pelos passos abaixos:

1. Para cada amostra supervisionada $(\mathbf{a}, \mathbf{b})$, construir $\mathbf{A}$ via codificação complementar (Eq. 4);
2. Selecionar uma categoria candidata **J** em $\mathrm{ART}_a$ por competição (Eqs. 8–9);
3. Verificar ressonância em $\mathrm{ART}_a$ pelo teste de vigilância (Eq. 10); se falhar, aplicar *reset* e repetir a busca;
4. Verificar consistência supervisionada no campo de mapeamento (Eq. 13); se falhar, acionar *match tracking* (Eq. 14) e repetir a busca em $\mathrm{ART}_a$;
5. Quando uma categoria **J** satisfaz simultaneamente vigilância e consistência supervisionada, atualizar $\mathbf{w}_J^a$ (Eq. 11) e $\mathbf{w}_J^{ab}$ (Eq. 15).

### 3.4.4 Predição

No modo de teste da rede, após selecionar uma categoria **J** em $\mathrm{ART}_a$, uma predição de classe pode ser obtida, por exemplo, por:

$$\hat{k} = \arg\max_{k \in \{1,\ldots,K\}} \left(w_{J,k}^{ab}\right), \tag{17}$$

onde $\mathbf{w}_{J,k}^{ab}$ é a **k**-ésima componente de $\mathbf{w}_J^{ab}$.

## 3.5 ANÁLISE INTERPRETATIVA DO USO DAS MATRIZES DE PESOS NA TRANSFERÊNCIA ENTRE DOMÍNIOS

Com a codificação complementar e o aprendizado por interseção, uma categoria pode ser interpretada como uma região de tolerância no hipercubo unitário: o protótipo $\mathbf{w}_j^a$ representa limites aprendidos por interseções sucessivas com entradas ressonantes (Carpenter; Grossberg;

Rosen, 1991). A vigilância regula a seletividade dessa região: valores maiores de $\rho_a$ induzem categorias mais específicas.

Em problemas com alta variabilidade intra-classe, ruído, sobreposição entre classes ou desalinhamentos de domínio, valores altos de vigilância combinados podem induzir proliferação de categorias e potencialmente prejudicar a generalização.

O objetivo dessa seção é discutir, de acordo com a teoria existente, o papel das matrizes FAM $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ no AT, interpretando o significado de transferí-las de um domínio de origem (ou fonte) para um domínio de destino (ou alvo).

### 3.5.1 O que representam $\mathbf{W}^a$ e $\mathbf{W}^{ab}$?

Após o treinamento em um domínio, suponha que $\mathrm{ART}_a$ tenha $\mathbf{C}$ categorias selecionadas. O conjunto de protótipos pode ser escrito por:

$$\mathbf{W}^a = \begin{bmatrix} (\mathbf{w}_1^a)^\top \\ (\mathbf{w}_2^a)^\top \\ \vdots \\ (\mathbf{w}_C^a)^\top \end{bmatrix} \in [0,1]^{C \times 2M}. \tag{18}$$

Cada linha de $\mathbf{W}^a$ é uma categoria/protótipo no espaço de entrada codificado, de modo que $\mathbf{W}^a$ armazena a partição/organização do espaço de atributos aprendida no domínio.

O campo de mapeamento pode ser escrito por:

$$\mathbf{W}^{ab} = \begin{bmatrix} (\mathbf{w}_1^{ab})^\top \\ (\mathbf{w}_2^{ab})^\top \\ \vdots \\ (\mathbf{w}_C^{ab})^\top \end{bmatrix} \in [0,1]^{C \times K}. \tag{19}$$

Cada linha de $\mathbf{W}^{ab}$ codifica o compromisso supervisionado da categoria $\mathbf{j}$ com o espaço de rótulos. No regime de aprendizado rápido ($\beta_{ab} = 1$), a atualização por interseção (Eq. 16) restringe quais rótulos são compatíveis com aquela categoria.

### 3.5.2 O que significa transferir $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ entre domínios?

No *AT* paramétrico (transferência de parâmetros), transferir um modelo significa inicializar o modelo do domínio-alvo com parâmetros aprendidos no domínio-fonte, antes do ajuste com dados do alvo (Pan; Yang, 2010). Na FAM, a transferência pode ser realizada por:

1. Transferir apenas $\mathbf{W}^a$: reutilizar protótipos como base de categorias no alvo, mas reaprender o mapeamento para rótulos (reinicializando ou liberando $\mathbf{W}^{ab}$). Essa escolha parece ser adequada, teoricamente, quando se admite que a geometria das classes no espaço de atributos é parcialmente reutilizável, mas que a relação atributo–classe pode variar.

2. Transferir $\mathbf{W}^a$ e $\mathbf{W}^{ab}$: reutilizar simultaneamente protótipos e associações de classe. Essa estratégia pode ser adequada, teoricamente, quando a tarefa é a mesma (mesmo conjunto de classes) e a mudança de domínio é predominantemente uma mudança em $p(\mathbf{x})$.

3. Transferir parcialmente $\mathbf{W}^a$: reutilizar apenas um subconjunto de protótipos (por exemplo, categorias mais frequentes ou mais estáveis) para reduzir risco de transferência negativa sob maior grau de desalinhamento entre domínios.

Após a transferência, a adaptação no domínio de destino ocorre por especialização e expansão incremental (mecanismo padrão da rede FAM). Categorias transferidas podem ser especializadas por interseção (Eq. 11) quando ressoam com padrões do alvo. Quando não há ressonância, novas categorias, exclusivas do alvo, são criadas, preservando as existentes. Se o mapeamento transferido $\mathbf{W}^{ab}$ induz erro supervisionado, o *match tracking* eleva $\rho_a$ e força subdivisões locais para recuperar consistência (Carpenter; Grossberg; Reynolds, 1991).

### 3.5.3 Resultados esperados sob diferentes cenários de transferência

Diferentes formas de *shift* (mudanças em $p(\mathbf{x})$, $p(y)$ ou $p(y|\mathbf{x})$) afetam o potencial de transferência (Ben-David *et al.*, 2010; Pan; Yang, 2010). O Quadro 1 apresenta comportamentos teoricamente esperados para a FAM, com base em sua dinâmica de busca, interseção e *match tracking*.

### 3.5.4 Discussão: por que a transferência pode funcionar (ou falhar) na FAM

A dinâmica da FAM pode favorecer a transferência quando domínios são razoavelmente alinhados, por dois motivos: reuso de cobertura do espaço via $\mathbf{W}^a$, levando a adaptação por especialização e correção orientada por erro via *match tracking*, que evita reescritas globais e induz subdivisões locais (Carpenter; Grossberg; Reynolds, 1991). Em contrapartida, sob desalinhamento severo, são esperados baixa ressonância generalizada e/ou mapeamento atributo-classe contraditório, aumentando o custo de busca, o número de categorias e o risco de transferência negativa.

## 3.6 USO DE FAM EM AT E DOMAIN ADAPTATION NA LITERATURA

Embora o termo *domain adaptation* nem sempre apareça explicitamente em trabalhos clássicos de ARTMAP, a família ART/ARTMAP foi concebida para aprendizado incremental em dados não estacionários. Revisões recentes sistematizam modelos ART e aplicações em engenharia, enfatizando propriedades de aprendizado contínuo e extensões para robustez (Carpenter *et al.*, 1992; Carpenter; Grossberg; Reynolds, 1991; Silva; Wunsch, 2019).

Na literatura consultada, não foi localizado o uso da FAM em AT da maneira proposta por esse trabalho, com transferências explícita de $\mathbf{W}^a$ e $\mathbf{W}^{ab}$, em diferentes combinações e estratégias.

Quadro 1 – Cenários de transferência de $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ na rede FAM e comportamento esperado.

| **Cenário de mudança** | **O que transferir** | **Comportamento esperado e observações** |
|---|---|---|
| *Covariate shift* (mesma tarefa, $p(\mathbf{x})$ muda) | $\mathbf{W}^a$ e $\mathbf{W}^{ab}$ | Se a mudança é moderada, protótipos transferidos tendem a ressonar e a se especializar para cobrir o alvo, reduzindo tempo de adaptação e o número de novas categorias. Se a mudança é forte (alvo fora do suporte), espera-se criação extensa de novas categorias, reduzindo o benefício de reuso. |
| *Label shift* ($p(y)$ muda) | $\mathbf{W}^a$ e (com cautela) $\mathbf{W}^{ab}$ | Protótipos podem permanecer úteis; entretanto, $\mathbf{W}^{ab}$ pode induzir viés para padrões dominantes da fonte. Ajustes no mapeamento e maior incidência de *match tracking* podem ser esperados ao reequilibrar decisões no alvo. |
| *Concept shift*/*conditional shift* ($p(y\|\mathbf{x})$ muda) | Preferencialmente $\mathbf{W}^a$ (ou nenhuma) | Transferir $\mathbf{W}^{ab}$ pode causar *negative transfer*. O *match tracking* tende a elevar $\rho_a$ e induzir subdivisões/criação de categorias para resolver inconsistências, aumentando $\mathbf{C}$ e potencialmente levando a sobreajuste se o alvo for pequeno. |
| Novas classes no alvo (expansão de $\mathbf{K}$) | $\mathbf{W}^a$ e expansão de $\mathbf{W}^{ab}$ | $\mathbf{W}^a$ pode acelerar reutilização para classes antigas (se presentes). Para novas classes, é necessário expandir $\mathbf{b}$ e as colunas de $\mathbf{W}^{ab}$; novas categorias podem então ser criadas e mapeadas às classes inéditas. |
| Mudança de representação (*features* diferentes) | Nenhuma, ou transferência parcial com alinhamento | Como $\mathbf{W}^a$ codifica limites no espaço de entrada, mudanças de escala/normalização/semântica reduzem ressonância e podem induzir proliferação de categorias. É necessário alinhamento/normalização (ou extrator estável) antes de transferir protótipos. |

Fonte: Autoria própria.

Foram localizados apenas trabalhos que utilizam a rede FAM em contextos que podem ser interpretados ou relacionados a *domain adaptation*. Alguns destes trabalhos e estratégias foram listados nas próximas subseções.

### 3.6.1 Transferência por reutilização de representação

Uma estratégia recorrente é empregar um extrator de características (potencialmente pré-treinado) e usar a FAM como classificador incremental no espaço de *features*. Nessa configuração, a transferência ocorre predominantemente pela representação do extrator, enquanto a FAM oferece treinamento rápido, incremental e capacidade de expansão por novas categorias. Em trabalhos acadêmicos, combinações entre aprendizado profundo (representação) e arquiteturas ART/ARTMAP têm sido exploradas em contextos de classificação incremental, coerentes com

taxonomias gerais de transferência por reutilização de representação (Nunes, 2024; Pan; Yang, 2010).

### 3.6.2 Aprendizado incremental e adaptação temporal

O arcabouço ARTMAP foi proposto explicitamente para aprendizado supervisionado em dados não estacionários. Estudos posteriores investigam treinamento incremental supervisionado com FAM, analisando desempenho e crescimento de categorias sob diferentes regimes de apresentação de dados. Esses cenários podem ser interpretados como adaptação temporal de domínio, pois a distribuição pode variar com o tempo e o modelo precisa incorporar novidades preservando conhecimento anterior (Carpenter; Grossberg; Reynolds, 1991; Connolly; Granger; Sabourin, 2008).

### 3.6.3 Extensões para robustez sob mudança de domínio

Extensões de FAM buscam controlar proliferação de categorias e melhorar generalização, frequentemente com impacto direto em robustez frente a variações de domínio. Um exemplo clássico é a introdução de pesos de entrada na FAM, que permite ponderar dimensões mais informativas e atenuar dimensões ruidosas. Revisões discutem ainda variantes com métricas adaptativas, estratégias de comitê e mecanismos de controle de complexidade, que podem ser interpretados como formas de melhorar estabilidade e adaptação sob *shift* (Andonie; Sasu, 2006; Silva; Wunsch, 2019).

### 3.6.4 Aplicações com condições variáveis e cenários abertos

Aplicações como reconhecimento em vigilância e cenários *open-set* envolvem variações de aquisição e padrões não observados no treinamento inicial. Há trabalhos que relatam uso de FAM em reconhecimento facial com poucas amostras por classe e regime *open-set*, onde indivíduos desconhecidos devem ser tratados de forma distinta (Al-Obaydy; Suandi, 2020). Esses cenários se conectam a *domain adaptation* na medida em que o sistema opera sob mudanças de distribuição e necessita expandir sua representação (novas categorias) sem degradar excessivamente as associações existentes.

### 3.6.5 Síntese

Em síntese, a literatura sugere que a FAM é especialmente atrativa quando há necessidade de aprendizado incremental/online; deseja-se treinamento rápido; a transferência ocorre via representação estável e o classificador precisa se adaptar continuamente. Em contrapartida, sob desalinhamento severo entre domínios, espera-se aumento e proliferação de categorias e risco de transferência negativa, demandando estratégias complementares como alinhamento/normalização, controle de relevâncias e seleção de protótipos transferidos (Andonie; Sasu, 2006; Silva; Wunsch, 2019).

# 4 MATERIAIS E MÉTODOS

De acordo com as definições clássicas de AT, o presente trabalho é melhor categorizado como AT (também chamado de adaptação de domínio), pois a tarefa preditiva é mantida fixa entre os domínios de origem e de destino (classificação binária de ocupação após transformação; mesma semântica de rótulos e mesmo objetivo), enquanto os domínios diferem devido a mudanças de sensor/configuração e, adicionalmente, devido ao deslocamento controlado introduzido no domínio alvo $T2(\lambda)$, que altera principalmente $P(X)$ (deslocamento de covariáveis). Sob a perspectiva do espaço de características, o problema é inicialmente heterogêneo (dimensionalidade/semânticas diferentes entre o domínio de origem $\mathbf{S}$ e o de destino $T1$), mas torna-se operacionalmente homogêneo após o alinhamento de características em um subespaço compartilhado: a transferência efetiva é então realizada nesse espaço alinhado. Em termos do mecanismo de transferência, melhor se enquadra em transferência de parâmetros/modelo ao reutilizar estruturas internas de conhecimento aprendidas no domínio de origem, especificamente as matrizes $\mathbf{W}^a$ (representações de categorias) e $\mathbf{W}^{ab}$ (associações categoria-classe) (Ben-David *et al.*, 2010; Pan; Yang, 2010; Weiss; Khoshgoftaar; Wang, 2016).

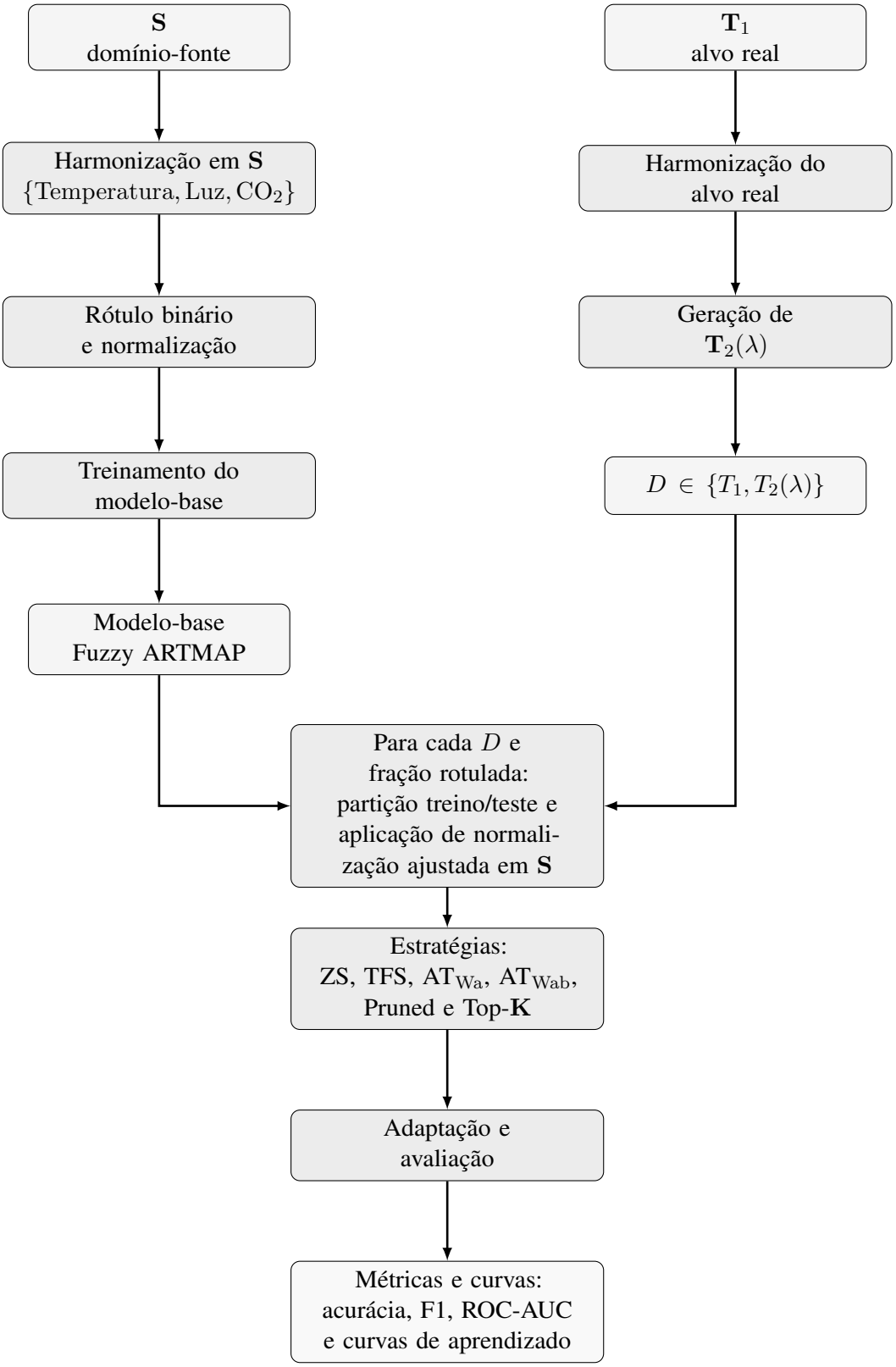
Este capítulo descreve as etapas executadas experimentalmente. A estratégia geral utilizada na metodologia é: treinar um modelo no domínio-fonte $\mathbf{S}$, aplicar diferentes estratégias de transferência de aprendizado (diretas e seletivas) para domínios-alvo $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$ (gerado a partir de deslocamento de $\mathbf{T}_1$), quantificar o desalinhamento entre domínios por métricas de distância (PAD, MMD e CORAL) e avaliar o desempenho final por métricas de classificação e curvas de aprendizado sob diferentes intensidades de *shift* (deslocamento) entre $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$.

A Figura 1 ilustra a visão geral do modelo de transferência proposto: a partir do conjunto fonte $\mathbf{S}$ realiza-se o treinamento inicial, aplicam-se estratégias de transferência para $\mathbf{T}_1$ e para o domínio sintético $\mathbf{T}_2(\lambda)$. Por fim, ainda, reportam-se métricas de desempenho e distâncias entre distribuições.

## 4.1 BASES DE DADOS UTILIZADAS

Nesta pesquisa, foram empregados dois bancos de dados públicos voltados à inferência de ocupação em ambientes internos: o *Occupancy Detection* e o *Room Occupancy Estimation*. Ambos pertencem à classe de dados multivariados com estrutura temporal e foram construídos a partir de sensores ambientais não intrusivos. Apesar dessa afinidade temática, os conjuntos diferem quanto à granularidade sensorial, à dimensionalidade do espaço de atributos e, em sua formulação original, à definição da variável-resposta. Tais diferenças tornam a combinação entre eles particularmente relevante para estudos de aprendizado por transferência, pois permitem investigar a reutilização de conhecimento entre domínios relacionados, mas não idênticos (Candanedo, 2016; Singh; Chaudhari, 2018).

Figura 1 – Visão geral do método: $S \rightarrow$ treino $\rightarrow$ transferência $\rightarrow \{T_1, T_2(\lambda)\}$.




Fonte: Autoria própria.

### 4.1.1 Occupancy Detection

O banco de dados *Occupancy Detection* (Candanedo, 2016) contém 20.560 instâncias e 6 atributos preditivos, além de um identificador e da variável-alvo (descrição no Quadro 2). A O conjunto não apresenta valores faltantes e foi construído para o problema de classificação binária da ocupação de uma sala. O rótulo de referência foi obtido a partir de imagens com marcação temporal registradas a cada minuto, o que confere maior confiabilidade à variável supervisionada. Em termos de modelagem, trata-se de uma base compacta e apropriada para estudos de classificação supervisionada, seleção de atributos e análise da contribuição relativa de sensores ambientais para a inferência da presença humana.

Quadro 2 – Variáveis do banco de dados *Occupancy Detection*.

| Variável | Papel | Descrição | Unidade/Formato |
|---|---|---|---|
| `id` | Identificador | Identificador numérico do registro. Em geral, não é tratado como atributo preditivo no treinamento dos modelos. | Sem unidade |
| `date` | Atributo | Instante temporal associado à observação. Pode ser explorado diretamente ou decomposto em atributos temporais derivados. | Data e hora no formato ano-mês-dia hora:minuto:segundo |
| `Temperature` | Atributo | Temperatura do ambiente interno no instante de medição. | °C |
| `Humidity` | Atributo | Umidade relativa do ar no ambiente monitorado. | % |
| `Light` | Atributo | Nível de iluminância medido no ambiente. | Lux |
| `CO2` | Atributo | Concentração de dióxido de carbono no ambiente. | ppm |
| `HumidityRatio` | Atributo | Razão de umidade, isto é, grandeza derivada da temperatura e da umidade relativa. | kg de vapor d'água/kg de ar |
| `Occupancy` | Variável-alvo | Estado de ocupação da sala. Assume valor 0 para ambiente não ocupado e valor 1 para ambiente ocupado. | Variável binária (0/1) |

Fonte: Elaborada com base em Candanedo (2016).

Observa-se que, nesse banco, os atributos físicos apresentam interpretação direta e unidades bem definidas, o que favorece a etapa de análise exploratória e a construção de modelos interpretáveis.

### 4.1.2 Room Occupancy Estimation

O banco de dados *Room Occupancy Estimation* (Singh; Chaudhari, 2018) contém 10.129 instâncias e 18 atributos preditivos, além da variável-alvo (descrição no Quadro 3). O experimento foi conduzido em uma sala de aproximadamente $6\,\mathrm{m} \times 4{,}6\,\mathrm{m}$, instrumentada com sete nós sensores e um nó de borda, com transmissão de dados a cada 30 segundos. A ocupação real foi anotada manualmente e variou entre 0 e 3 pessoas ao longo da coleta. Em sua formulação original, trata-se de uma base voltada à estimação da contagem de ocupantes, o que a torna mais rica em termos de heterogeneidade sensorial e mais desafiadora sob o ponto de vista de AM.

Quadro 3 – Variáveis do banco de dados *Room Occupancy Estimation*.

| Variável | Papel | Descrição | Unidade/Formato |
|---|---|---|---|
| `Date` | Atributo | Data de aquisição da observação. | YYYY/MM/DD |
| `Time` | Atributo | Horário de aquisição da observação. | HH:MM:SS |
| `S1_Temp` | Atributo | Temperatura medida no nó sensor S1. | °C |
| `S2_Temp` | Atributo | Temperatura medida no nó sensor S2. | °C |
| `S3_Temp` | Atributo | Temperatura medida no nó sensor S3. | °C |
| `S4_Temp` | Atributo | Temperatura medida no nó sensor S4. | °C |
| `S1_Light` | Atributo | Iluminância medida no nó sensor S1. | Lux |
| `S2_Light` | Atributo | Iluminância medida no nó sensor S2. | Lux |
| `S3_Light` | Atributo | Iluminância medida no nó sensor S3. | Lux |
| `S4_Light` | Atributo | Iluminância medida no nó sensor S4. | Lux |
| `S1_Sound` | Atributo | Sinal associado ao sensor de som no nó S1; corresponde à saída analógica do amplificador lida pelo conversor A/D do microcontrolador. | Volts |
| `S2_Sound` | Atributo | Sinal associado ao sensor de som no nó S2; corresponde à saída analógica do amplificador lida pelo conversor A/D do microcontrolador. | Volts |
| `S3_Sound` | Atributo | Sinal associado ao sensor de som no nó S3; corresponde à saída analógica do amplificador lida pelo conversor A/D do microcontrolador. | Volts |
| `S4_Sound` | Atributo | Sinal associado ao sensor de som no nó S4; corresponde à saída analógica do amplificador lida pelo conversor A/D do microcontrolador. | Volts |
| `S5_CO2` | Atributo | Concentração de dióxido de carbono medida no nó sensor S5. | ppm |
| `S5_CO2_Slope` | Atributo | Inclinação temporal dos valores de $CO_2$ calculada em uma janela deslizante. | Unidade não explicitada no repositório |
| `S6_PIR` | Atributo | Indicador binário de detecção de movimento proveniente do sensor PIR no nó S6. | Variável binária |
| `S7_PIR` | Atributo | Indicador binário de detecção de movimento proveniente do sensor PIR no nó S7. | Variável binária |
| `Room_Occupancy_Count` | Variável-alvo | Número real de ocupantes presentes na sala no instante de medição; corresponde ao *ground truth* anotado manualmente. | Contagem discreta de ocupantes |

Fonte: Elaborada com base em Singh e Chaudhari (2018).

### 4.1.3 Análise do uso das bases em aprendizado por transferência

No presente trabalho, o banco *Room Occupancy Estimation* foi utilizado como domínio fonte e o banco *Occupancy Detection* como domínio alvo. Entretanto, antes da transferência, a tarefa originalmente multiclasse do domínio fonte foi reduzida a uma formulação binária de ocupação. Com essa transformação, a variável-resposta do domínio fonte passou a representar, de forma análoga ao domínio alvo, a distinção entre ambiente ocupado e não ocupado. Essa decisão reduz a discrepância semântica entre as tarefas, de modo que a transferência deixa de ocorrer entre uma tarefa de contagem e uma tarefa de detecção, passando a ocorrer entre dois problemas de classificação binária relacionados. Sob a ótica metodológica, essa adequação é importante porque diminui o descompasso entre os espaços de decisão e torna mais plausível a reutilização de representações aprendidas no domínio fonte.

Além da binarização da variável-alvo no domínio fonte, a transferência foi conduzida utilizando-se apenas o subconjunto comum de variáveis entre as duas bases. Em termos práticos, isso significa que o processo de adaptação foi restrito às modalidades sensoriais compartilhadas pelos domínios, particularmente aquelas associadas à temperatura, à iluminância e à concentração de $CO_2$, excluindo-se variáveis sem correspondência direta entre os conjuntos. Dessa forma, atributos presentes apenas no domínio fonte, como som e sensores PIR, bem como atributos

específicos do domínio alvo, como umidade relativa e razão de umidade, não participaram diretamente da transferência. Essa escolha é metodologicamente consistente, pois reduz o desajuste entre os espaços de entrada e evita que o processo de transferência dependa de informações indisponíveis em um dos domínios.

Sob essa configuração, a transferência passa a avaliar se regularidades físicas associadas à presença humana, aprendidas no domínio fonte por meio de variáveis ambientalmente correlatas, permanecem úteis quando o modelo é adaptado ao domínio alvo. Em outras palavras, o foco deixa de estar na riqueza sensorial exclusiva do *Room Occupancy Estimation* e passa a recair sobre a parcela do conhecimento potencialmente invariável entre os domínios. Tal escolha fortalece a coerência experimental, pois a comparação entre as bases passa a ser realizada em um espaço observacional compatível, no qual a transferência é sustentada por sinais que possuem interpretação física análoga nos dois conjuntos.

Mesmo com essa harmonização, a transferência não se torna trivial. Persistem diferenças importantes entre os domínios, como a instrumentação empregada, a resolução temporal das coletas, a forma de distribuição espacial dos sensores e as características específicas dos ambientes monitorados. Enquanto o *Room Occupancy Estimation* foi coletado em uma sala instrumentada com múltiplos nós sensores e transmissão a cada 30 segundos, o *Occupancy Detection* foi organizado a partir de medições ambientais agregadas e rótulos obtidos por imagens temporizadas a cada minuto. Essas diferenças indicam que, mesmo após o alinhamento das tarefas e do subconjunto de atributos, ainda subsiste uma mudança de domínio no sentido estatístico, isto é, as distribuições marginais e conjuntas das variáveis podem não coincidir entre a base fonte e a base alvo.

Do ponto de vista do aprendizado por transferência, essa configuração apresenta uma vantagem metodológica clara: ela reduz simultaneamente duas fontes clássicas de transferência negativa, a saber, a incompatibilidade entre tarefas e a incompatibilidade entre atributos. Ao binarizar a saída do domínio fonte, o estudo aproxima a estrutura supervisonada dos dois problemas; ao restringir a entrada ao subconjunto comum de variáveis, aproxima os espaços de observação. Como consequência, o desempenho obtido no domínio alvo tende a refletir de forma mais direta a capacidade de reutilização de padrões ambientais efetivamente compartilhados entre as bases, e não o efeito de discrepâncias artificiais introduzidas por diferenças de rotulagem ou por atributos inexistentes em um dos domínios.

Em síntese, o uso do *Room Occupancy Estimation* como domínio fonte e do *Occupancy Detection* como domínio alvo, após a binarização do problema fonte e a seleção do subconjunto comum de variáveis, configura uma estratégia experimental conceitualmente consistente para avaliar aprendizado por transferência em inferência de ocupação. Nessa formulação, investiga-se se padrões latentes associados à dinâmica ambiental da presença humana podem ser aprendidos em um domínio relacionado e posteriormente reutilizados em outro contexto observacional. Assim, a contribuição da transferência passa a ser analisada em condições de maior comparabilidade estrutural, sem eliminar, contudo, os desafios reais impostos pela mudança de distribuição

entre os domínios.

## 4.2 DEFINIÇÃO DOS DOMÍNIOS

O experimento é composto por três domínios: um domínio-fonte $\mathbf{S}$ e dois domínios-alvo, $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$. O domínio $\mathbf{T}_1$ corresponde ao conjunto-alvo "real", enquanto $\mathbf{T}_2(\lambda)$ é um domínio sintético que controla a severidade do deslocamento por meio do parâmetro $\lambda \in [0, 1]$.

### 4.2.1 Domínio-fonte $\mathbf{S}$

O conjunto $\mathbf{S}$ (*Room Occupancy Estimation*) foi obtido do repositório público *UCI Machine Learning Repository*, (Singh; Chaudhari, 2018), cuja descrição detalhada foi apresentada na seção 4.1.2.

Após o carregamento do dataset aplicou-se:

1. extração de atributos e rótulos $(X, y)$;
2. seleção de subconjunto de atributos e padronização de colunas, para selecionar as variáveis ambientais análogas ao domínio-alvo (*Temperature*, *Light* e *CO2*);
3. ajuste no período de amostragem dos dados (de 30 s para 1 minuto, para ser compatível com a amostragem dos dados de $\mathbf{T}_1$);
4. codificação do rótulo (reescrevendo todas as saídas diferentes de zero como 1 para reduzir à mesma tarefa de $\mathbf{T}_1$ ) e filtragem de colunas não informativas.

### 4.2.2 Domínio-alvo $\mathbf{T}_1$

O conjunto de dados *Occupancy Detection*, utilizado como um dos domínios alvo, também foi obtido do repositório público *UCI Machine Learning Repository*, (Candanedo, 2016), e a descrição detalhada foi apresentada na seção 4.1.1.

O conjunto $\mathbf{T}_1$ corresponde a um dos domínio-alvo ou domínio de destino. Após o carregamento, é feita uma projeção para o mesmo espaço de atributos de $\mathbf{S}$ (seção 4.3) e executado o particionamento treino/teste com estratificação por classe. Essa separação é reutilizada em todas as estratégias, de modo que a comparação seja feita sob os mesmos conjuntos de teste.

### 4.2.3 Domínio-alvo sintético $\mathbf{T}_2(\lambda)$

O domínio $\mathbf{T}_2(\lambda)$ é construído a partir de $\mathbf{T}_1$ com diferentes intensidade de deslocamento (*shift*), representado por $(\lambda)$. A construção de $\mathbf{T}_2(\lambda)$ é feita em duas etapas:

1. geração de uma versão deslocada $\mathbf{T}_1^{\text{shift}}$ por transformações paramétricas aplicadas aos atributos (Seção 4.5);

2. combinação convexa entre $\mathbf{T}_1$ e $\mathbf{T}_1^{\text{shift}}$ controlada por $\lambda$:

$$\mathbf{x}_{T_2(\lambda)} = (1-\lambda)\,\mathbf{x}_{T_1} + \lambda\,\mathbf{x}_{T_1^{\text{shift}}}, \quad \lambda \in [0,1]. \tag{20}$$

Quando $\lambda = 0$, tem-se $T_2(\lambda) = T_1$ (ausência de deslocamento); quando $\lambda = 1$, tem-se o deslocamento máximo.

A Eq. (20) define uma família contínua de domínios intermediários ao interpolar, de forma convexa, entre $\mathbf{T}_1$ e uma versão deslocada $\mathbf{T}_1^{\text{shift}}$, permitindo controlar a severidade do deslocamento por $\lambda \in [0,1]$. Estratégias baseadas em interpolação convexa são amplamente empregadas para gerar amostras sintéticas e regularizar modelos e também para analisar robustez/detectar *shift* por meio da interpolação entre instâncias provenientes de domínios distintos (por exemplo, em *Domain Interpolation Sensitivity*). Além disso, em generalização de domínio, a região de "interpolação" entre ambientes é frequentemente modelada como combinações convexas das distribuições observadas (Gokhale; Feinglass; Yang, 2022; Rosenfeld; Ravikumar; Risteski, 2022; Zhang *et al.*, 2018).

## 4.3 HARMONIZAÇÃO DE ATRIBUTOS

Como $\mathbf{S}$ e $\mathbf{T}_1$ possuem conjuntos de atributos distintos, a transferência só é bem definida após a projeção para um espaço de atributos compartilhado. Essa etapa é realizada por:

1. padronização dos nomes de colunas (por exemplo, normalização de strings e harmonização de sufixos);
2. definição do conjunto de atributos comuns $\mathcal{F} = \mathcal{F}_S \cap \mathcal{F}_{T_1}$;
3. filtragem de ambas as matrizes de atributos para manter apenas $\mathcal{F}$, resultando em matrizes $\mathbf{X}_S \in \mathbb{R}^{n_S \times d}$ e $\mathbf{X}_{T_1} \in \mathbb{R}^{n_T \times d}$, com o mesmo $d = |\mathcal{F}|$.

Essa harmonização reduz o risco de *shift* artificial causado por atributos inexistentes em um dos domínios e viabiliza o cálculo direto de distâncias PAD, MMD e CORAL sobre o mesmo espaço $\mathbb{R}^d$.

## 4.4 PRÉ-PROCESSAMENTO

O mesmo pré-processamento é aplicado em $\mathbf{S}$, $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$, visando garantir comparabilidade numérica entre atributos e estabilizar o aprendizado. Foram aplicados os procedimentos das subseções seguintes.

### 4.4.1 Normalização e consistência

A normalização, do tipo *min–max*, foi realizada atributo a atributo. Para cada dimensão $\mathbf{j}$ do vetor de entrada, estimam-se os extremos observados no conjunto de referência,

$$m_j = \min_i x_{i,j}, \qquad M_j = \max_i x_{i,j}, \tag{21}$$

definindo-se a amplitude $r_j = M_j - m_j$. A transformação é então dada por

$$\tilde{x}_{i,j} = \frac{x_{i,j} - m_j}{r_j}. \tag{22}$$

Para garantir estabilidade numérica, adota-se uma salvaguarda quando $r_j = 0$ (isto é, quando o atributo é constante no conjunto de referência): nesses casos, impõe-se uma amplitude efetiva estritamente positiva, por exemplo $r_j \leftarrow 1$ (equivalentemente, $M_j \leftarrow m_j + 1$), evitando-se divisão por zero e assegurando uma transformação bem definida.

Após a reescala, pode-se aplicar uma projeção para o intervalo unitário, de modo a respeitar o domínio esperado pelo modelo subsequente:

$$\tilde{x}_{i,j} \leftarrow \min\{1, \max\{0, \tilde{x}_{i,j}\}\}.$$

Como checagens de consistência, verifica-se:

- se os parâmetros de normalização foram previamente estimados no conjunto de referência;
- a dimensionalidade do vetor a ser transformado é compatível com a dimensionalidade utilizada na estimação dos extremos;
- se as saídas permanecem numéricas e finitas (isto é, livres de `NaN` e `Inf`), além de satisfazerem $\tilde{x}_{i,j} \in [0, 1]$ quando a projeção é empregada.

### 4.4.2 Particionamento treino/teste e amostragem por fração

Para gerar curvas de aprendizado, o conjunto de treino do domínio-alvo é subamostrado em frações $f$ (Seção 4.10), estratificadas por classe, preservando a proporção das classes no treino e no teste. Em todas as repetições, o conjunto de teste permanece fixo, de modo que variações observadas reflitam apenas o efeito de $f$ e da estratégia de transferência empregada.

A fim de assegurar reprodutibilidade, adotou-se o controle explícito da semente aleatória (*seed*). Em cada execução, a semente foi aplicada às rotinas pseudoaleatórias relevantes (embaralhamento/particionamento dos dados, inicialização de parâmetros), garantindo que o mesmo protocolo experimental pudesse ser reproduzido. Além disso, as métricas de desempenho foram estimadas a partir de $N$ ($N = 5$) execuções independentes com sementes distintas, e os resultados são reportados como média $\pm$ desvio-padrão, o que fornece uma medida de robustez frente à aleatoriedade do processo.

## 4.5 GERAÇÃO DE *SHIFT* E CONSTRUÇÃO DE $\mathbf{T}_2(\lambda)$

O algoritmo de geração de *shift* aplica a transformação em espaço normalizado $[0,1]^d$, com parâmetros específicos para três grupos de atributos identificados por padrão de nome:

- atributos contendo *Temperature* (grupo $\mathcal{G}_T$);
- atributos contendo *Light* (grupo $\mathcal{G}_L$);
- atributos contendo *CO2* (grupo $\mathcal{G}_C$).

Atributos fora desses grupos são mantidos inalterados e não são utilizados para compor o $\mathbf{T}_2(\lambda)$. O treinamento ocorre, nos três domínios, com os mesmos atributos, por escolha de escopo do trabalho e por requisito da rede FAM.

Seja $\mathbf{x}$ um atributo normalizado em $[0,1]$, com média estimada $\mu$. A transformação paramétrica (aplicada elemento a elemento) tem a forma:

$$\tilde{x} = \mathrm{clip}\left((1+g)\,x - g\,\mu + b + \varepsilon,\ 0,\ 1\right), \tag{23}$$

em que $\mathbf{b}$ é um termo de deslocamento aditivo (*bias*), $\mathbf{g}$ é um termo de escala (*gain*), $\varepsilon \sim \mathcal{N}(0,\sigma^2)$ representa ruído gaussiano e $\mathrm{clip}(\cdot)$ limita o intervalo a $[0,1]$ (Bishop, 1995). No procedimento experimental realizado, $(\mathbf{b},\mathbf{g},\sigma)$ são especificados por grupo ($\mathbf{T}$, $L$ e $\mathbf{C}$), conforme Tabela 4. Após gerar $\mathbf{T}_1^{\text{shift}}$, constrói-se $\mathbf{T}_2(\lambda)$ segundo a Eq. (20).

Tabela 4 – Parâmetros fixados para a geração do *shift* em $\mathbf{T}_2(\lambda)$.

| **Parâmetro** | **Valor** |
|---|---|
| $b_T$ (*bias* em Temperatura) | 0,80 |
| $g_L$ (*gain* em Luminosidade) | 0,80 |
| $b_C$ (*bias* em $CO_2$) | 0,60 |
| $\sigma_T$ (ruído em Temperatura) | 0,02 |
| $\sigma_L$ (ruído em Luminosidade) | 0,02 |
| $\sigma_C$ (ruído em $CO_2$) | 0,03 |

Fonte: Autoria própria.

No experimento, adotou-se uma perturbação seletiva por grupo de atributos. Para o grupo de Temperatura, utilizou-se deslocamento aditivo e ruído, com $\mathbf{g}_T = 0$; para o grupo de Luminosidade, utilizou-se transformação de escala em torno da média e ruído, com $\mathbf{b}_L = 0$; para o grupo de $CO_2$, utilizou-se deslocamento aditivo e ruído, com $\mathbf{g}_C = 0$. Assim, os parâmetros não utilizados em cada grupo foram fixados em zero. Essa escolha é compatível com a literatura de calibração e deriva de sensores, na qual efeitos aditivos e multiplicativos podem ser modelados separadamente, e modelos simplificados podem reter apenas os termos pertinentes ao fenômeno de interesse (Balzano; Nowak, 2007; Wang *et al.*, 2015).

## 4.6 TREINAMENTO NO DOMÍNIO-FONTE $\mathbf{S}$

O treinamento inicial no domínio-fonte $\mathbf{S}$ produz um modelo-base a partir do qual são derivadas as estratégias de transferência. Em termos gerais o procedimento executado é:

1. construção do conjunto de treino em $\mathbf{S}$ (com controle de semente de aleatoriedade e estratificação quando aplicável);
2. ajuste do classificador com os hiperparâmetros da FAM;
3. armazenamento do "estado" treinado (parâmetros e estruturas internas necessárias à inferência), que será reutilizado na etapa de transferência.

## 4.7 MEDIÇÃO DE DISTÂNCIAS ENTRE DOMÍNIOS

Para quantificar o desalinhamento/distância entre $\mathbf{S}$ e os domínios-alvo, bem como entre $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$, foram utilizadas três medidas: *PAD (Proxy A-Distance), MMD (Maximum Mean Discrepancy) e CORAL (Correlation Alignment)*, com formulações consolidadas em adaptação de domínios (Ben-David *et al.*, 2006; Ben-David *et al.*, 2010; Ganin *et al.*, 2016; Gretton *et al.*, 2007; Gretton *et al.*, 2012; Sun; Feng; Saenko, 2017; Sun; Saenko, 2016). Na descrição e apresentação das métricas, nas subseções 4.7.1, 4.7.2 e 4.7.3, $\mathbf{S}$ e $\mathbf{T}$ denotam genericamente o par de domínios sob comparação. No procedimento experimental realizado, as combinações de domínios são $(S, T) \in \{(S, T_1), (S, T_2(\lambda)), (T_1, T_2(\lambda))\}$.

### 4.7.1 Proxy A-Distance (PAD)

A *Proxy A-Distance* (PAD) é uma estimativa prática da divergência $d_{\mathcal{A}}(S, T)$ entre dois domínios $\mathbf{S}$ e $\mathbf{T}$, obtida a partir da capacidade de um classificador binário de distinguir de qual domínio uma amostra provém. Constrói-se um conjunto rotulado artificialmente com $\mathbf{z} = 0$ para amostras de $\mathbf{S}$ e $\mathbf{z} = 1$ para amostras de $\mathbf{T}$ e treina-se um classificador de domínio $g_{S/T} : \mathcal{X} \to \{0, 1\}$. Denotando por $\hat{\epsilon}_{S/T}$ o erro empírico de $\mathbf{g}_{S/T}$ (e, portanto, $\mathrm{Acc}_{S/T} = 1 - \hat{\epsilon}_{S/T}$), define-se:

$$\mathrm{PAD}(S, T) = 2\left(1 - 2\hat{\epsilon}_{S/T}\right) = 4\,\mathrm{Acc}_{S/T} - 2, \tag{24}$$

de modo que $\mathrm{PAD}(S, T) \in [0, 2]$: valores próximos de $0$ indicam que os domínios são pouco distinguíveis ($\mathrm{Acc}_{S/T} \approx 0{,}5$), enquanto valores próximos de $2$ indicam forte desalinhamento (classificador quase perfeito). A forma de (24) é a aproximação clássica baseada no erro do discriminador de domínios (Ben-David *et al.*, 2006; Ben-David *et al.*, 2010; Ganin *et al.*, 2016).

### 4.7.2 Maximum Mean Discrepancy (MMD)

A *Maximum Mean Discrepancy* (MMD) mede a distância entre distribuições comparando as médias de suas representações em um espaço de Hilbert reproduzível (RKHS) $\mathcal{H}_k$ induzido por

um *kernel* $k(\cdot,\cdot)$. Seja $\phi(\cdot)$ o mapeamento de características associado. A definição populacional é:

$$\mathrm{MMD}(S,T) = \left\|\mathbb{E}_{\mathbf{x}\sim S}[\phi(\mathbf{x})] - \mathbb{E}_{\mathbf{y}\sim T}[\phi(\mathbf{y})]\right\|_{\mathcal{H}_k}. \tag{25}$$

Na prática, dada uma amostra $\{\mathbf{x}_i\}_{i=1}^{n}$ de $\mathbf{S}$ e $\{\mathbf{y}_j\}_{j=1}^{m}$ de $\mathbf{T}$, uma estimativa não viesada de $\mathrm{MMD}^2(S,T)$ é dada por:

$$\mathrm{MMD}^2(S,T) = \frac{1}{n(n-1)}\sum_{i\neq i'} k(\mathbf{x}_i,\mathbf{x}_{i'}) + \frac{1}{m(m-1)}\sum_{j\neq j'} k(\mathbf{y}_j,\mathbf{y}_{j'}) - \frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} k(\mathbf{x}_i,\mathbf{y}_j). \tag{26}$$

Valores maiores indicam maior discrepância entre distribuições; a escolha do *kernel* controla quais diferenças estatísticas são enfatizadas (Gretton *et al.*, 2007; Gretton *et al.*, 2012).

### 4.7.3 CORAL (Correlation Alignment)

A métrica CORAL (*Correlation Alignment*) quantifica o desalinhamento entre domínios comparando estatísticas de segunda ordem (covariâncias) das representações. Seja $\mathbf{X}_S \in \mathbb{R}^{n\times d}$ a matriz de atributos do domínio $\mathbf{S}$ (linhas correspondem às amostras) após centralização por média, e analogamente $\mathbf{X}_T \in \mathbb{R}^{m\times d}$. As covariâncias amostrais são definidas por:

$$\mathbf{C}_S = \frac{1}{n-1}\mathbf{X}_S^{\top}\mathbf{X}_S, \qquad \mathbf{C}_T = \frac{1}{m-1}\mathbf{X}_T^{\top}\mathbf{X}_T. \tag{27}$$

A distância CORAL é então dada por:

$$\mathrm{CORAL}(S,T) = \frac{1}{4d^2}\left\|\mathbf{C}_S - \mathbf{C}_T\right\|_F^2, \tag{28}$$

em que $\|\cdot\|_F$ é a norma de Frobenius e $\mathbf{d}$ é o número de atributos. Valores maiores indicam maior discrepância de covariância (e, portanto, maior desalinhamento estrutural) entre domínios (Sun; Saenko, 2016; Sun; Feng; Saenko, 2017).

## 4.8 ESTRATÉGIAS DE TRANSFERÊNCIA DE APRENDIZADO

As estratégias foram implementadas como variações controladas do que é transferido do modelo treinado em $\mathbf{S}$ e do que é readaptado no domínio-alvo. A descrição abaixo enfatiza o procedimento experimental executado e a motivação de cada estratégia. Notações internas específicas da FAM (e suas equações de aprendizado) são omitidas, pois já foram apresentadas no Capítulo 3.

### 4.8.1 Baselines

Duas estratégias baselines foram executadas, *zero-shot* e *train from scratch*.

#### 4.8.1.1 ZS (*Zero-Shot*)

Na estratégia ZS, o modelo é treinado exclusivamente em $\mathbf{S}$ e avaliado diretamente em $\mathbf{T}_1$ e/ou $\mathbf{T}_2(\lambda)$ sem qualquer atualização no domínio-alvo. O procedimento é:

1. treinar em $\mathbf{S}$ com o protocolo padrão;
2. aplicar inferência no conjunto de teste de $\mathbf{T}_1$ (ou $\mathbf{T}_2(\lambda)$) e computar as métricas.

O objetivo dessa baseline é quantificar a degradação de desempenho causada pelo *shift* quando não há adaptação.

#### 4.8.1.2 TFS (*Train From Scratch*)

Na estratégia TFS, treina-se um modelo do zero no domínio-alvo, usando apenas dados do próprio domínio. O algoritmo repete o treinamento para diferentes frações de dados rotulados do alvo (Seção 4.10), obtendo curvas de aprendizado. O procedimento é:

1. amostrar uma fração $f$ do conjunto de treino do domínio-alvo (estratificada por classe);
2. treinar o modelo com inicialização padrão (sem reutilizar parâmetros de $\mathbf{S}$);
3. avaliar no conjunto de teste do domínio-alvo.

A baseline *Train From Scratch* (TFS) fornece uma referência sem transferência, permitindo quantificar o ganho (ou transferência negativa) das abordagens de AT ao comparar seu desempenho com o treinamento do modelo nos domínios-alvo a partir do zero, sob o mesmo protocolo experimental.

### 4.8.2 Transferência direta com *fine-tuning* controlado

As estratégias $\mathrm{AT_{Wa}}$e $\mathrm{AT_{Wab}}$transferem o estado treinado em $\mathbf{S}$ para inicializar o treinamento no domínio-alvo e então realizam readaptação parcial (*fine-tuning*) com dados do alvo. A diferença entre elas é o subconjunto de parâmetros internos readaptados: a variante $\mathrm{AT_{Wa}}$restringe a atualização a parâmetros associados à representação de entrada, enquanto $\mathrm{AT_{Wab}}$libera também parâmetros associados às conexões/associações com o rótulo.

#### 4.8.2.1 $\mathrm{AT_{Wa}}$(adaptação restrita)

1. treinar o modelo em $\mathbf{S}$ e salvar seu estado;
2. inicializar um novo treinamento no domínio-alvo carregando o estado de $\mathbf{S}$;
3. liberar atualização apenas do subconjunto de parâmetros de representação (“Wa”) durante o treino em $\mathbf{T}_1$ ou $\mathbf{T}_2(\lambda)$;
4. avaliar no teste do domínio-alvo.

#### 4.8.2.2 $\mathrm{AT}_{\mathrm{Wab}}$(adaptação ampliada)

1. treinar o modelo em $\mathbf{S}$ e salvar seu estado;
2. inicializar o treinamento no domínio-alvo com o estado de $\mathbf{S}$;
3. liberar atualização tanto dos parâmetros de representação ("Wa") quanto das associações com rótulos ("Wab");
4. avaliar no teste do domínio-alvo.

### 4.8.3 Transferência Seletiva (Top-K) (Selective Transfer Tuning (Top-K))

As estratégias *Top*-$\mathbf{K}$ (ou *Selective Transfer Tuning*) procuram mitigar a transferência negativa evitando transferir estruturas internas do domínio-fonte que não são úteis nos domínios-alvo. O algoritmo seleciona um subconjunto de categorias/unidades aprendidas em $\mathbf{S}$ com base em estatísticas obtidas ao aplicar o modelo de $\mathbf{S}$ sobre dados de treino do alvo.

O procedimento geral é:

1. executar inferência do modelo treinado em $\mathbf{S}$ sobre um conjunto de treino do alvo, registrando: a categoria vencedora/ativada e o rótulo verdadeiro;
2. computar, para cada categoria $\mathbf{c}$, uma medida de relevância (vide Equação (29));
3. selecionar o conjunto $\mathcal{C}_K$ com as $\mathbf{K}$ categorias mais relevantes (vide Equações (30) e (31));
4. inicializar o modelo-alvo somente com parâmetros relativos a $\mathcal{C}_K$ e realizar *fine-tuning* no domínio-alvo.

A medida de relevância para cada categoria $\mathbf{j}$ foi calculada a partir do protótipo armazenado na linha $\mathbf{j}$ da matriz de pesos do módulo ART ($\mathbf{w}_j \in \mathbb{R}^d$, em que $\mathbf{d}$ é a dimensionalidade do espaço de entrada após normalização/codificação). A relevância é definida como a massa do protótipo, dada pela norma $L_1$:

$$r_j \;=\; \|\mathbf{w}_j\|_1 \;=\; \sum_{p=1}^{d} |w_{j,p}| \,. \tag{29}$$

Como os pesos da FAM são não negativos, i.e., após normalização e/ou codificação de complemento, a Eq. (29) reduz-se, na prática, à soma dos componentes do protótipo $\mathbf{w}_j$, intuitivamente, categorias com maior $\mathbf{r}_j$ correspondem a protótipos com maior "massa" acumulada e, portanto, são priorizadas na transferência seletiva.

Em relação ao procedimento de seleção de categorias mais relevantes, após computar $\{r_j\}_{j=1}^{N_{\mathrm{cat}}}$ para as $\mathbf{N}_{\mathrm{cat}}$ categorias existentes no modelo-fonte, as categorias são ordenadas em ordem

decrescente de relevância. Define-se $\mathbf{K}_{\text{eff}}$ como o valor efetivo de $\mathbf{K}$ limitado ao intervalo válido (ao menos uma categoria e no máximo o total):

$$K_{\text{eff}} \;=\; \min(\max(1, K),\; N_{\text{cat}})\,. \tag{30}$$

Em seguida, seleciona-se o conjunto de índices das $\mathbf{K}_{\text{eff}}$ categorias com maiores valores de $\mathbf{r}_j$:

$$\mathcal{K} \;=\; \mathrm{TopK}\big(\{r_j\}_{j=1}^{N_{\text{cat}}}\big) \;=\; \{j \;:\; r_j \text{ está entre os } K_{\text{eff}} \text{ maiores valores}\}\,. \tag{31}$$

O procedimento mantém exatamente $\mathbf{K}_{\text{eff}}$ categorias após a ordenação por $\mathbf{r}_j$, descartando as demais na etapa de transferência seletiva. Os resultados experimentais indicaram maior eficiência para valores pequenos de $\mathbf{K}$, então, por se tratar de classificação binária, foi utilizado $K = 2$ no algoritmo principal e uma varredura $topk - grid = (1, 2, 3, 5, 10)$ foi usada na etapa de *tuning*.

Foram avaliadas duas variantes dessa estratégia:

- TOPK$_{\text{Wa}}$: seleção Top-$\mathbf{K}$ e readaptação restrita ("Wa");
- TOPK$_{\text{Wab}}$: seleção Top-$\mathbf{K}$ e readaptação ampliada ("Wa" + "Wab").

### 4.8.4 Poda por pureza

A estratégia PRUNED aplica uma poda (*pruning*), ou corte, após a transferência, removendo categorias transferidas que exibem baixa "pureza" no domínio-alvo. A pureza de uma categoria é definida como o grau de dominância de uma classe entre as amostras do alvo que a ativam.

Seja $\mathcal{I}_c$ o conjunto de índices de amostras do alvo cuja categoria vencedora é $\mathbf{c}$, e $\mathbf{n}_{c,k}$ o número de amostras em $\mathcal{I}_c$ pertencentes à classe $\mathbf{k}$. Define-se:

$$\mathrm{purity}(c) = \frac{\max_k n_{c,k}}{\sum_k n_{c,k}} \in [0, 1]. \tag{32}$$

Dado um limiar $\tau \in (0, 1]$, mantém-se apenas categorias com $\mathrm{purity}(c) \geq \tau$. No conjunto de execuções de *tuning*, foram avaliados $\tau \in \{0{,}70,\ 0{,}80,\ 0{,}85,\ 0{,}90\}$.

O procedimento *pruned* executado é:

1. treinar em $\mathbf{S}$ e transferir o estado para inicialização no alvo;
2. executar inferência do modelo transferido sobre um conjunto de treino do alvo para identificar a categoria vencedora por amostra;
3. calcular $\mathrm{purity}(c)$ por categoria via Eq. (32);
4. remover/inativar categorias com pureza abaixo de $\tau$ antes do *fine-tuning*;
5. realizar *fine-tuning* no domínio-alvo e avaliar no teste.

Foram avaliadas duas variantes dessa estratégia:

- PRUNED$_{\mathrm{Wa}}$: *pruning* + readaptação restrita (“Wa”);
- PRUNED$_{\mathrm{WaWab}}$: *pruning* + readaptação ampliada (“Wa” + “Wab”).

## 4.9 ANÁLISE DE SENSIBILIDADE AOS HIPERPARÂMETROS

Foi realizada uma análise de sensibilidade dos hiperparâmetros do classificador. A Tabela 5 resume os valores inspecionados para os parâmetros $\rho$ e $\beta$, bem como para intensidades de *shift* $\lambda$ utilizadas na sensibilidade. Para cada combinação, o desempenho é calculado no domínio-alvo sob o mesmo protocolo de avaliação descrito na Seção 4.11.

Tabela 5 – Grade de valores utilizada na análise de sensibilidade (*grid search*) dos hiperparâmetros e do nível de *shift*.

| **Parâmetro** | **Valores** |
|---|---|
| $\lambda$ (nível de *shift* em $T_2(\lambda)$) | $\{0{,}00;\ 0{,}50;\ 1{,}00\}$ |
| $\rho$ (vigilância) | $\{0{,}70;\ 0{,}80;\ 0{,}85;\ 0{,}90;\ 0{,}95\}$ |
| $\beta$ (taxa de aprendizado) | $\{0{,}50;\ 0{,}80;\ 1{,}00\}$ |

Fonte: Autoria própria.

## 4.10 PROTOCOLO EXPERIMENTAL E CURVAS DE APRENDIZADO

A fim de caracterizar o impacto do tamanho da amostra rotulada do domínio-alvo, foram geradas curvas de aprendizado variando a fração de treino $f$ do domínio-alvo. As frações utilizadas foram

$$f \in \{0{,}05;\ 0{,}10;\ 0{,}20;\ 0{,}30;\ 0{,}50;\ 0{,}70;\ 1{,}00\}. \tag{33}$$

Para cada combinação de domínio-alvo, intensidade de *shift* $\lambda$ e fração $f$, o experimento é repetido com múltiplas inicializações/amostragens aleatórias e os resultados são agregados como média e desvio-padrão. Nas curvas de aprendizado reportadas, foram utilizadas cinco repetições por ponto de curva.

## 4.11 MÉTRICAS DE AVALIAÇÃO

O desempenho da classificação é reportado por métricas complementares:

1. **Acurácia**:

$$\mathrm{Acc} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\left(\hat{y}_i = y_i\right); \tag{34}$$

2. **F1-score**:

$$\mathrm{F1}_k = \frac{2\,\mathrm{Prec}_k\,\mathrm{Rec}_k}{\mathrm{Prec}_k + \mathrm{Rec}_k}, \quad \mathrm{F1}_{\text{macro}} = \frac{1}{K}\sum_{k=1}^{K}\mathrm{F1}_k; \tag{35}$$

3. **AUC-ROC**.

   A métrica ROC–AUC foi estimada a partir de um escore contínuo derivado diretamente da medida de *match* do módulo $A$ da rede *Fuzzy ARTMAP* no cenário binário. Para cada amostra $\mathbf{x}$ (após a preparação interna do modelo, incluindo a codificação por complemento quando aplicável), calculou-se o grau de casamento com cada categoria $\mathbf{j}$ por

$$m_j(\mathbf{x}) \;=\; \frac{\sum_k \min(x_k,\, w_{j,k})}{\sum_k x_k + \varepsilon}, \qquad \varepsilon > 0, \tag{36}$$

   em que $\mathbf{w}_j$ denota o vetor de pesos (protótipo) associado à categoria $\mathbf{j}$. Cada categoria $\mathbf{j}$ está associada a um rótulo predito $\widehat{c}(j) \in \{0, 1\}$. Em seguida, definiu-se um escore discriminativo para a classe positiva como a diferença entre o maior *match* dentre as categorias associadas à classe $1$ e o maior *match* dentre as categorias associadas à classe $0$:

$$s(\mathbf{x}) \;=\; \max_{j:\,\widehat{c}(j)=1} m_j(\mathbf{x}) \;-\; \max_{j:\,\widehat{c}(j)=0} m_j(\mathbf{x}). \tag{37}$$

   Esse escore não corresponde a uma probabilidade calibrada e sim a uma medida contínua e reprodutível de evidência relativa para a classe positiva, apropriada para ordenação e, consequentemente, para a construção da curva ROC. Finalmente, a ROC–AUC é computada a partir dos pares $\{(y_i, s_i)\}_{i=1}^{n}$, isto é,

$$\text{ROC-AUC} \;=\; \mathrm{AUC}\big(\mathrm{ROC}\big(\{y_i\}, \{s_i\}\big)\big)\,, \tag{38}$$

   sendo a AUC considerada indefinida quando o conjunto avaliado contém apenas uma classe.

## 4.12 CONSIDERAÇÕES DE IMPLEMENTAÇÃO E REPRODUTIBILIDADE

A execução foi realizada com controle da semente de aleatoriedade nas rotinas de amostragem e particionamento. As etapas de pré-processamento, geração de $\mathbf{T}_2(\lambda)$ e avaliação são aplicadas de forma idêntica entre as estratégias, garantindo que diferenças observadas sejam atribuíveis às decisões de transferência.

# 5 RESULTADOS E DISCUSSÃO

Este capítulo apresenta e discute os resultados experimentais obtidos, divididos pelos seguintes tópicos: caracterização do desalinhamento entre domínios por meio de métricas de distância, desempenho preditivo da FAM sob as diferentes estratégias de transferência propostas, sensibilidade aos parâmetros de vigilância ($\rho$) e taxa de aprendizado ($\beta$), efeito do grau de deslocamento entre domínios controlado por $\lambda$ (isto é, a severidade da transformação aplicada na construção de $\mathbf{T}_2(\lambda)$ a partir de $\mathbf{T}_1$), efeito do regime de dados no domínio de destino (fração de amostras usadas no treino) e as estratégias de transferência testadas.

As estratégias avaliadas foram: TFS (treinamento do zero no alvo), ZS (zero-shot) e variantes de transferência na FAM com inicialização/seleção de pesos ($\text{AT}_{\text{Wa}}$, $\text{AT}_{\text{Wab}}$, $\text{PRUNED}_{\text{Wa}}$, $\text{PRUNED}_{\text{WaWab}}$, $\text{TOPK}_{\text{Wa}}$, $\text{TOPK}_{\text{Wab}}$). As métricas reportadas são média e desvio padrão sobre as repetições do protocolo experimental: acurácia, $F_1$, AUC-ROC e número médio de categorias (*cat*) formadas na camada competitiva da rede FAM.

## 5.1 DISTÂNCIAS ENTRE DOMÍNIOS E EFEITO DE $\lambda$

A Tabela 6 apresenta um resumo das medidas de distância entre os domínios: PAD (*proxy A-distance*), MMD (RBF) e CORAL. O objetivo é verificar se o parâmetro $\lambda$ induz, de fato, um aumento sistemático do afastamento entre $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$ e também o efeito do *shift* em relação ao domínio-fonte ($\mathbf{S}$).

Tabela 6 – Métricas de distância entre domínios ($S, T_1$ e $T_2(\lambda)$) em função de $\lambda$.

| $\lambda$ | PAD ($T_1$–$T_2(\lambda)$) | MMD ($T_1$–$T_2(\lambda)$) | CORAL ($T_1$–$T_2(\lambda)$) | PAD (S–$T_2(\lambda)$) | MMD (S–$T_2(\lambda)$) | CORAL (S–$T_2(\lambda)$) |
|---|---|---|---|---|---|---|
| 0,00 | -0,0253 | 0,0000 | 0,0000000 | 1,9973 | 0,2753 | 0,0008533 |
| 0,25 | -0,0013 | 0,0007 | 0,0000002 | 1,9987 | 0,2888 | 0,0008366 |
| 0,50 | 0,0253 | 0,0029 | 0,0000009 | 1,9987 | 0,3030 | 0,0008203 |
| 0,75 | 0,0693 | 0,0066 | 0,0000020 | 2,0000 | 0,3176 | 0,0008045 |
| 1,00 | 0,1413 | 0,0117 | 0,0000035 | 2,0000 | 0,3330 | 0,0007886 |

Fonte: Autoria própria.

Antes da análise geral dos resultados das métricas obtidas, cabe realizar uma observação sobre os valores de PAD ($\mathbf{T}_1$–$\mathbf{T}_2(\lambda)$) em $\lambda = 0,00$ e $0,25$, que resultaram em sinal negativo. Estimativas negativas de PAD podem ocorrer quando o discriminador de domínios obtém acurácia $< 0,5$ devido à variância amostral/instabilidade do ajuste. Como o problema é simétrico, utiliza-se $acc' = \max(acc, 1 - acc)$, garantindo $PAD >= 0$. Valores negativos de pequena magnitude, como os obtidos na configuração em questão, indicam desempenho próximo do acaso e, portanto, discrepância próxima de zero (Schat *et al.*, 2020).

Puderam ser observadas duas tendências principais nas métricas de distâncias entre $\mathbf{S}$, $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$ :

- Aumento do afastamento $\mathbf{T}_1$–$\mathbf{T}_2(\lambda)$ com $\lambda$: a MMD ($\mathbf{T}_1$–$\mathbf{T}_2(\lambda)$) cresce monotonicamente de 0 até $\approx 0,0117$, e CORAL ($\mathbf{T}_1$–$\mathbf{T}_2(\lambda)$) também aumenta (embora em escala muito pequena). Isso indica que o mecanismo de geração de $\mathbf{T}_2(\lambda)$ introduz deslocamento progressivo, conforme planejado na metodologia;

- Deslocamento em relação ao domínio de origem ($\mathbf{S}$): a PAD (S–$\mathbf{T}_2(\lambda)$) permanece muito alta (próxima de 2) para todos os $\lambda$, sugerindo forte separabilidade entre $\mathbf{S}$ e os domínios de destino após o pré-processamento adotado. A MMD ($S - T_2(\lambda)$) aumenta de $\approx 0,275$ para $\approx 0,333$ com $\lambda$, indicando aumento de discrepância. Já a CORAL ($S - T_2(\lambda)$) reduz levemente, dado que seus valores absolutos são da ordem de $10^{-3}$. Esse efeito pode ser interpretado como pequeno e pode indicar que o *shift* é mais dominado por diferenças de distribuição não capturadas por covariância, ou que a escala do dado torna CORAL pouco sensível nesse aspecto.

Relacionando a distância entre domínios com a teoria ARTMAP, a alta discrepância entre domínios tende a aumentar a ocorrência de *mismatch* com categorias previamente aprendidas, elevando a probabilidade de *reset* e criação de novas categorias. Assim, espera-se que estratégias de transferência que inicializam protótipos ($\mathbf{W}^a$) possam tanto favorecer (se as categorias forem reutilizáveis) quanto prejudicar (se induzirem proliferação de categorias).

## 5.2 RESULTADOS NO DOMÍNIO ALVO ($\mathbf{T}_1$)

A Tabela 7 resume o desempenho no domínio de destino $\mathbf{T}_1$, no regime supervisionado completo, ou seja com todos os dados disponíveis (fração $= 1,0$). As Figura 2, Figura 3, Figura 4 e Figura 5 apresentam um resumo comparativo das estratégias para $frac = 1$, com número de categorias ao longo de $\lambda$, por estratégia.

Em seguida, discute-se o comportamento ao longo das frações de dados rotulados e a relação desempenho–complexidade (número de categorias formadas).

### 5.2.1 Principais resultados em $\mathbf{T}_1$

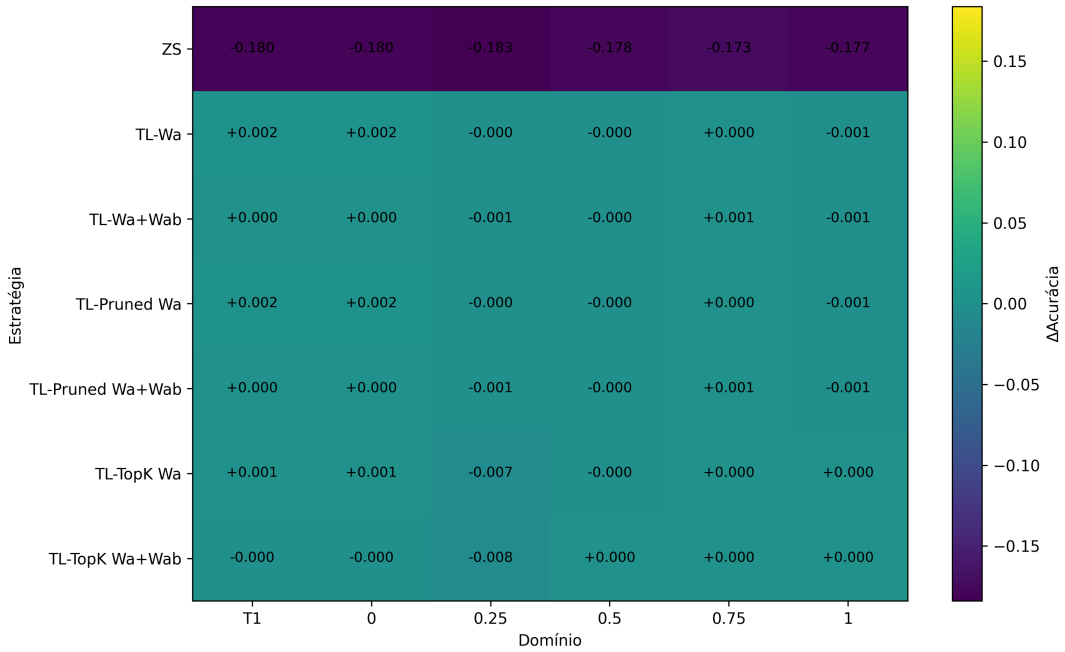
- TFS já é muito forte em $\mathbf{T}_1$, dificultando ganhos significativos de transferência: o treinamento do zero (TFS) em $\mathbf{T}_1$ alcança acurácia média $\approx 0,944$, com variação na ordem de $0,6$–$0,8$ p.p. (desvio-padrão). As variantes de transferência obtêm ganhos marginais em relação a acurácia: o melhor caso ($\text{AT}_{\text{Wa}}$/$\text{PRUNED}_{\text{Wa}}$) eleva a acurácia em $0,1556$ p.p., um incremento pequeno frente à variabilidade do protocolo. Isso sugere que, para os dados usados e com a metodologia proposta, a FAM consegue se adaptar eficientemente a $\mathbf{T}_1$ sem depender criticamente de inicialização transferida.

Tabela 7 – Desempenho em $\mathbf{T}_1$ (fração $= 1,0$). Média $\pm$ desvio padrão.

| **Estratégia** | **Acurácia** | $F_1$ | **AUC-ROC** | **Categorias** |
|---|---|---|---|---|
| TFS | 0,9444 $\pm$ 0,0075 | 0,8745 $\pm$ 0,0196 | 0,9134 $\pm$ 0,0447 | 283 $\pm$ 185 |
| $\text{AT}_{\text{Wa}}$(= $\text{PRUNED}_{\text{Wa}}$) | 0,9460 $\pm$ 0,0062 | 0,8784 $\pm$ 0,0165 | 0,8959 $\pm$ 0,0698 | 338 $\pm$ 184 |
| $\text{AT}_{\text{Wab}}$(= $\text{PRUNED}_{\text{WaWab}}$) | 0,9447 $\pm$ 0,0069 | 0,8751 $\pm$ 0,0184 | 0,9180 $\pm$ 0,0516 | 339 $\pm$ 187 |
| $\text{TOPK}_{\text{Wa}}$ | 0,9456 $\pm$ 0,0066 | 0,8776 $\pm$ 0,0173 | 0,8976 $\pm$ 0,0732 | 285$\pm$ 185 |
| $\text{TOPK}_{\text{Wab}}$ | 0,9444 $\pm$ 0,0074 | 0,8745 $\pm$ 0,0193 | 0,9128 $\pm$ 0,0452 | 285$\pm$ 185 |
| ZS | 0,7645 $\pm$ 0,0003 | 0,0000 $\pm$ 0,0000 | 0,9809 $\pm$ 0,0008 | 56 $\pm$ 0,0 |

Fonte: Autoria própria.

Figura 2 – Resumo (frac = 1): $\Delta$Acc em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia.

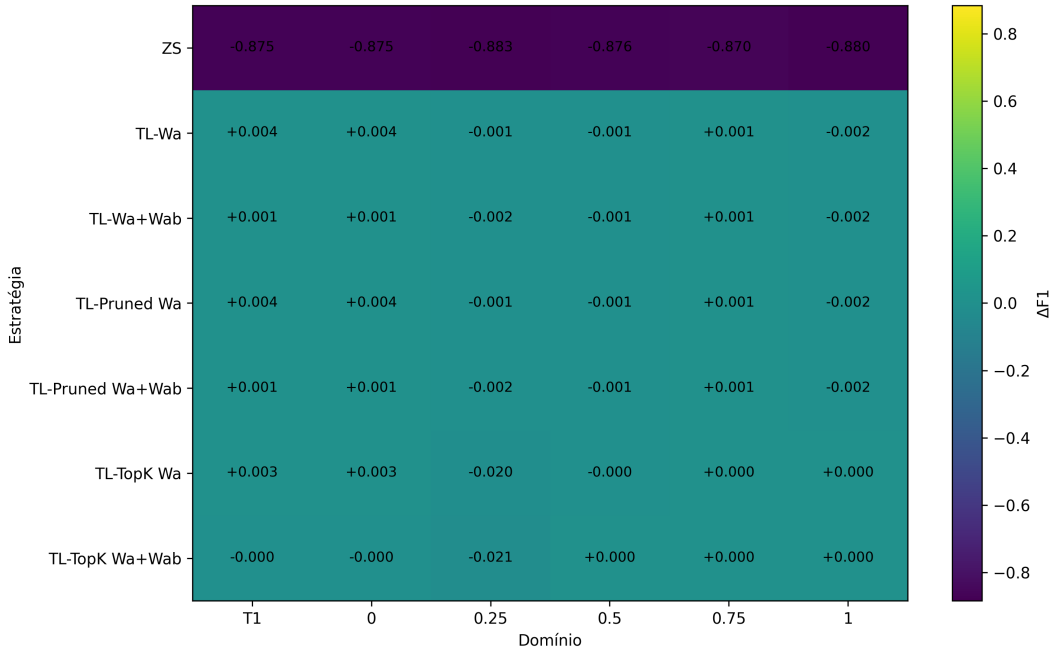



Fonte: Autoria própria.

- A transferência via $\mathbf{W}^a$ aumenta a complexidade do modelo: em $\mathbf{T}_1$ com fração $= 1,0$, TFS forma em média $283$ categorias, enquanto $\text{AT}_{\text{Wa}}$forma cerca de $339$ categorias. Na teoria ART/ARTMAP, esse aumento é consistente com o mecanismo de vigilância e *match tracking*: protótipos transferidos podem não cobrir adequadamente regiões do espaço do domínio alvo, elevando a frequência de *reset* e criação de novas categorias para manter consistência supervisionada (Carpenter *et al.*, 1992; Carpenter; Grossberg; Rosen, 1991). Porém, o aumento do número de categorias criadas não influenciou negativamente na acurácia do modelo.

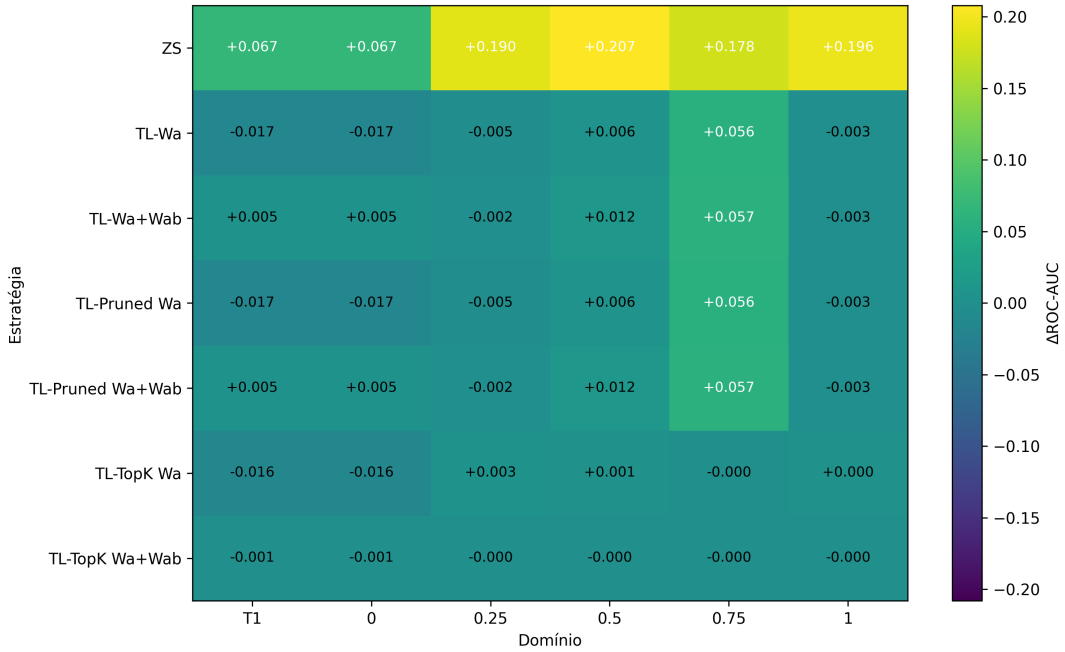
- Zero-shot falha na classe positiva ($F1 = 0$), apesar de AUC-ROC elevado: a estratégia ZS apresenta acurácia $\approx 0,764$ e $F_1 = 0$, indicando colapso para a classe majoritária, comportamento compatível com desalinhamento entre as categorias/mapeamento aprendidos.

Figura 3 – Resumo (frac = 1): $\Delta F_1$ em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia.




Fonte: Autoria própria.

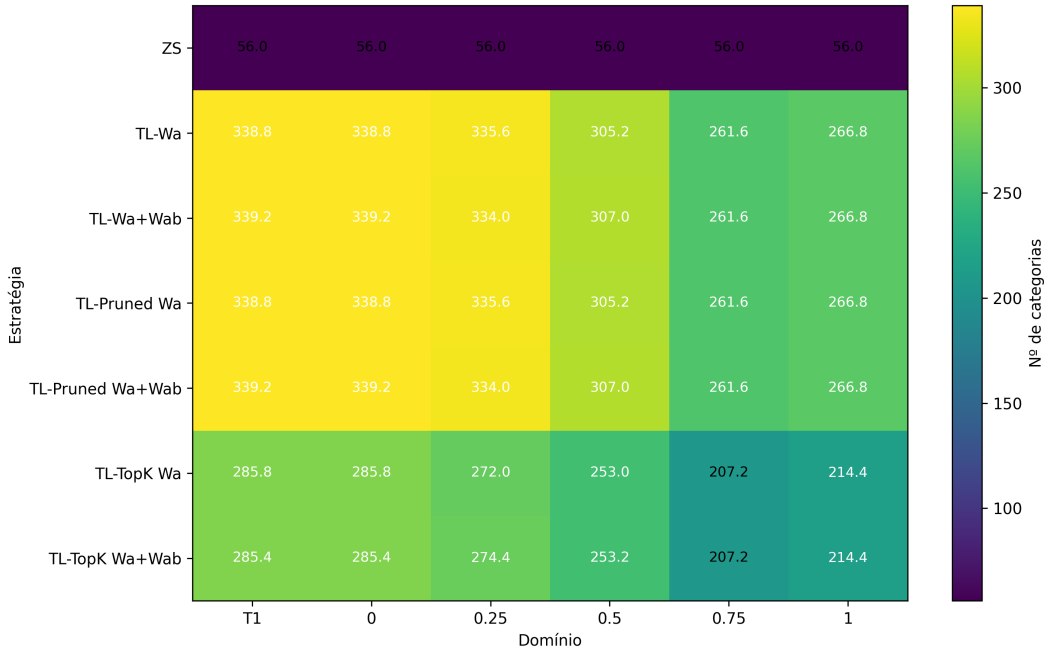
Figura 4 – Resumo (frac = 1): $\Delta$AUC-ROC em relação a $\mathbf{T}_1$ ao longo de $\lambda$, por estratégia.




Fonte: Autoria própria.

- A estratégia TopK reduz a proliferação de categorias mantendo o desempenho: $\textsc{TopK}_{\text{Wa}}$ preserva o número de categorias muito próximo ao TFS e mantém acurácia comparável. Isso sugere que a seleção de um subconjunto de protótipos transferidos (TopK) atua como regularizador estrutural: reduz complexidade adicional sem penalizar o desempenho.

Figura 5 – Resumo (frac = 1): número de categorias ao longo de $\lambda$, por estratégia.




Fonte: Autoria própria.

### 5.2.2 Efeito da fração de dados rotulados empregada

As Figura 6, Figura 7, Figura 8, Figura 9 ilustram as curvas de aprendizado obtidas no domínio $T_1$ (acurácia, $F_1$, AUC-ROC e número de categorias) em função da fração (particionamento) dos dados rotulados utilizada no treinamento ($frac = 0,05; 0,10; 0,20; 0,30; 0,50; 0,70$ e $1,00$).

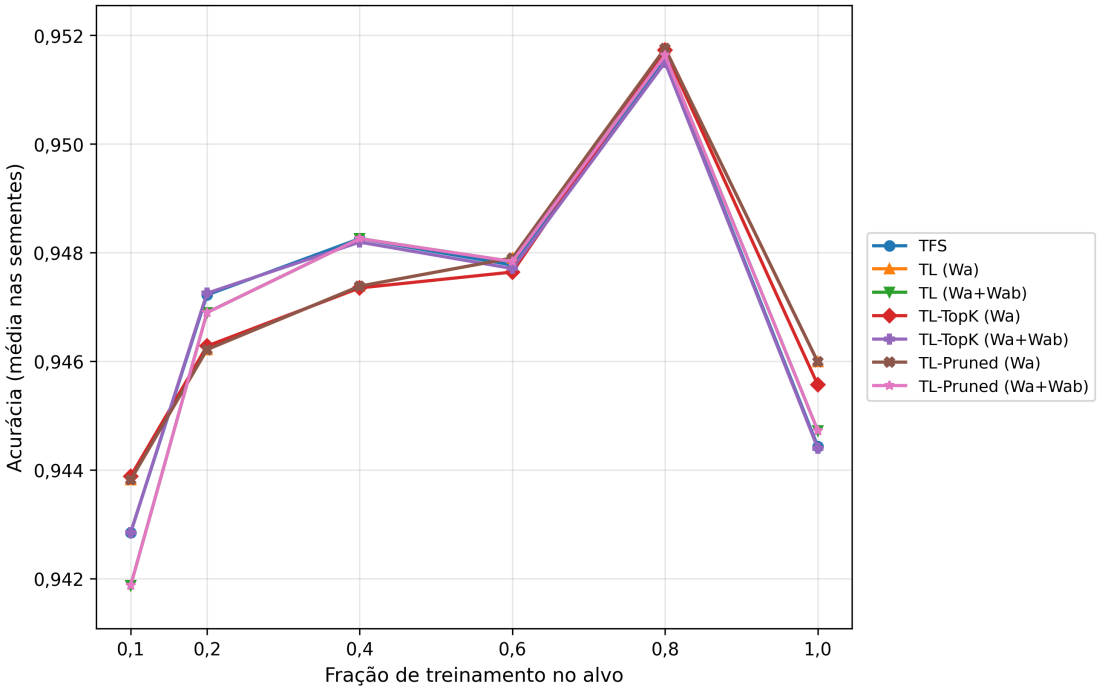
A acurácia permanece alta e relativamente estável (tipicamente $0,94 - 0,95$) para TFS e para as variantes de transferência, mesmo em frações pequenas, sugerindo que o problema em $\mathbf{T}_1$ possui sinal discriminativo forte ou que o pré-processamento utilizado e a dinâmica da FAM foram suficientes para rápida adaptação.

O número de categorias cresce substancialmente com o aumento da fração, refletindo a sensibilidade da FAM à ordem de apresentação e ao regime de *fast learning*. A transferência via $W_a$ eleva o patamar de categorias já em frações pequenas (por exemplo, em 0,05: TFS $\approx 33$ vs $\text{AT}_{\text{Wa}} \approx 88$), sem ganhos proporcionais em acurácia. Isso reforça a interpretação do *trade-off* desempenho-complexidade existente.

## 5.3 RESULTADOS NO DOMÍNIO ALVO ($\mathbf{T}_2(\lambda)$)

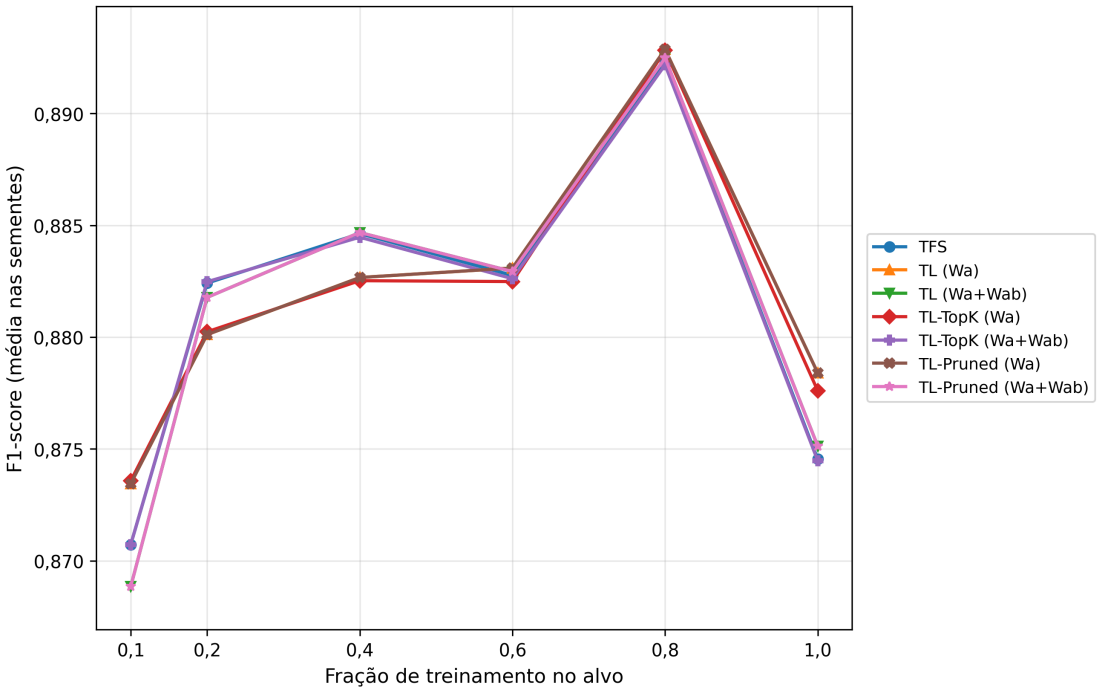
Para sintetizar o efeito do (deslocamento) adicionado, as Tabelas 8 e 9 mostram, no regime de fração $= 1,0$, a variação de acurácia de cada método em relação ao baseline TFS e o número médio de categorias formadas. Para uma análise mais detalhada das curvas de aprendizado em $\mathbf{T}_2(\lambda)$ são apresentadas, também, as Tabelas 10, 11, 12 e 13, que, para $\lambda = 0,25$ e $\lambda = 1,0$ trazem a acurácia e F-1 média por fração de dados.

Figura 6 – Curvas de aprendizado em $\mathbf{T}_1$: acurácia em função da fração de dados rotulados (`frac`).



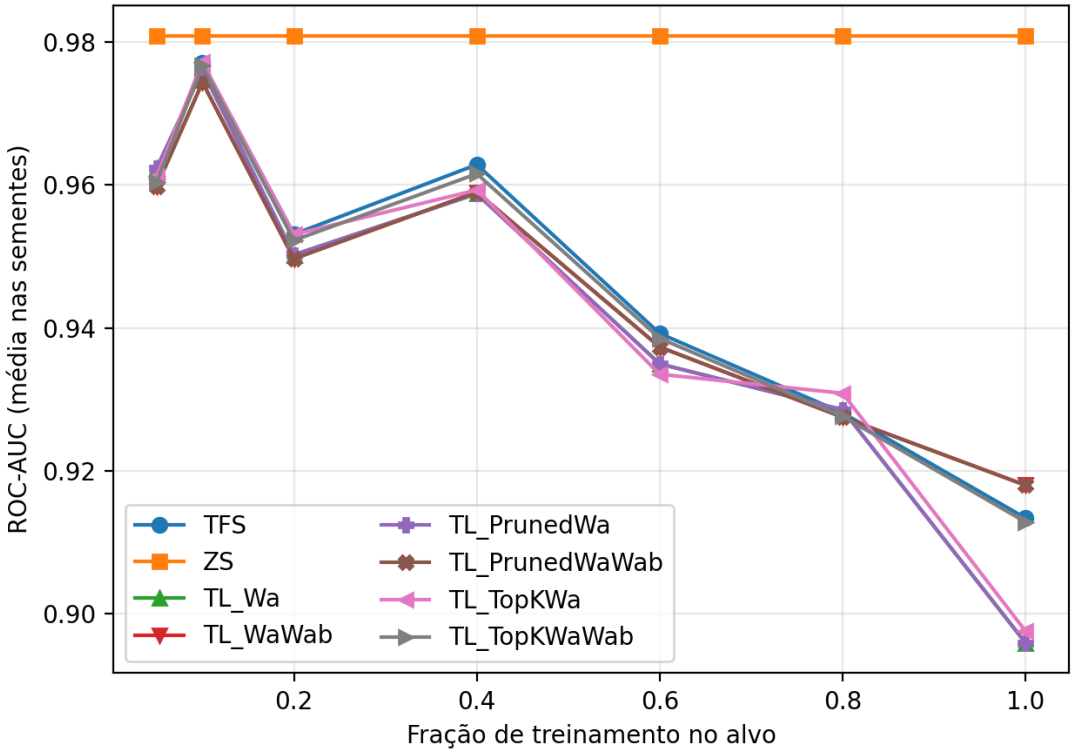

Fonte: Autoria própria.

Figura 7 – Curvas de aprendizado em $\mathbf{T}_1$: $F_1$ em função da fração de dados rotulados (`frac`).




Fonte: Autoria própria.

Figura 8 – Curvas de aprendizado em $\mathbf{T}_1$: AUC-ROC em função da fração de dados rotulados (`frac`).




Fonte: Autoria própria.

Tabela 8 – Variação de acurácia (em pontos percentuais) relativa ao TFS no regime fração $= 1,0$.

| Estratégia | $\mathbf{T}_1$ | $\lambda = 0$ | 0,25 | 0,50 | 0,75 | 1,00 |
|---|---|---|---|---|---|---|
| ZS | -18,00 | -18,00 | -18,31 | -17,79 | -17,34 | -17,69 |
| $\text{AT}_{\text{Wa}}$ | +0,16 | +0,16 | -0,05 | -0,03 | +0,03 | -0,06 |
| $\text{AT}_{\text{Wab}}$ | +0,03 | +0,03 | -0,09 | -0,02 | +0,06 | -0,06 |
| $\text{PRUNED}_{\text{Wa}}$ | +0,16 | +0,16 | -0,05 | -0,03 | +0,03 | -0,06 |
| $\text{PRUNED}_{\text{WaWab}}$ | +0,03 | +0,03 | -0,09 | -0,02 | +0,06 | -0,06 |
| $\text{TOPK}_{\text{Wa}}$ | +0,11 | +0,11 | -0,73 | -0,01 | +0,00 | +0,00 |
| $\text{TOPK}_{\text{Wab}}$ | -0,00 | -0,00 | -0,76 | +0,00 | +0,00 | +0,00 |

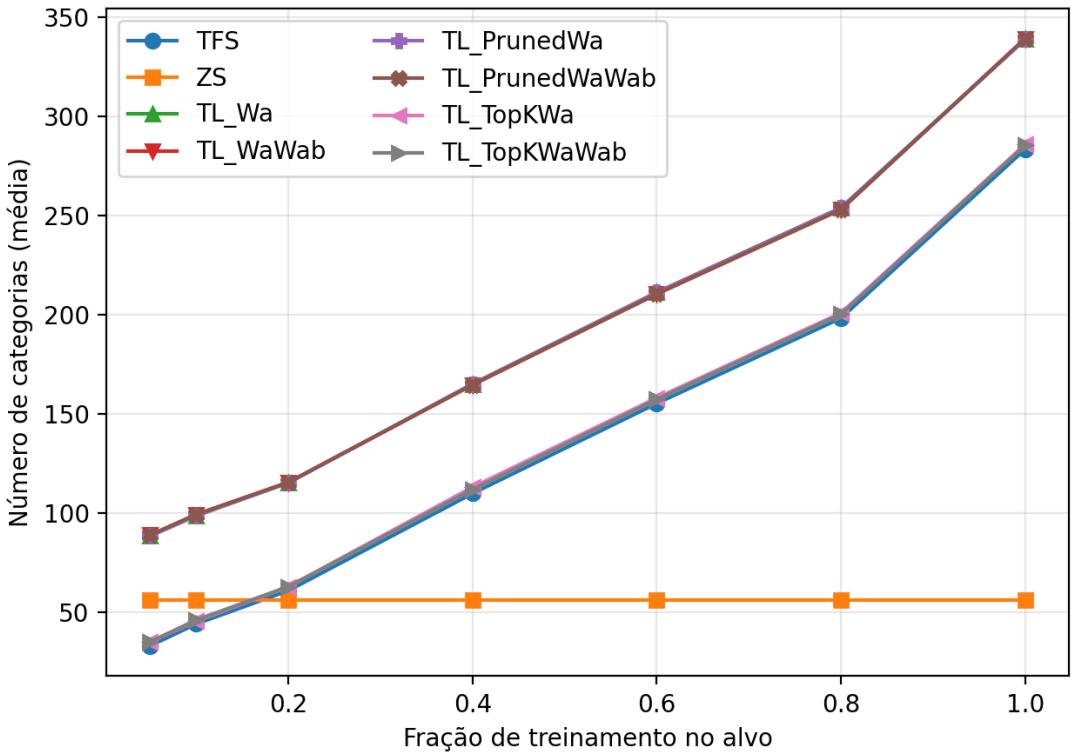
Fonte: Autoria própria.

### 5.3.1 Robustez da acurácia da FAM

A partir dos resultados obtidos para ($\mathbf{T}_2(\lambda)$), podem ser feitas duas observações, a respeito da robustez ao aumento do deslocamento, do crescimento do número de categorias da FAM, e do melhor desempenho conjunto em *shifts* moderados.

Mesmo com o aumento do deslocamento as variações de acurácia entre estratégias permanecem pequenas, sugerindo que a rede FAM mantêm a consistência no aprendizado mesmo sob efeito dos diferentes níveis de deslocamento induzidos. Também deve ser observado que o baseline TFS é competitivo e que a tarefa não depende fortemente de reutilização de conhecimento.

Figura 9 – Curvas de aprendizado em $\mathbf{T}_1$: número de categorias em função da fração de dados rotulados (`frac`).




Fonte: Autoria própria.

Tabela 9 – Número médio de categorias (*cat*) no regime fração $= 1,0$.

| Estratégia | $\mathbf{T}_1$ | $\lambda = 0$ | 0,25 | 0,50 | 0,75 | 1,00 |
|---|---|---|---|---|---|---|
| ZS | 56,0 | 56,0 | 56,0 | 56,0 | 56,0 | 56,0 |
| $\text{AT}_{\text{Wa}}$ | 338,8 | 338,8 | 335,6 | 305,2 | 261,6 | 266,8 |
| $\text{AT}_{\text{Wab}}$ | 339,2 | 339,2 | 334,0 | 307,0 | 261,6 | 266,8 |
| $\text{PRUNED}_{\text{Wa}}$ | 338,8 | 338,8 | 335,6 | 305,2 | 261,6 | 266,8 |
| $\text{PRUNED}_{\text{WaWab}}$ | 339,2 | 339,2 | 334,0 | 307,0 | 261,6 | 266,8 |
| $\text{TOPK}_{\text{Wa}}$ | 285,8 | 285,8 | 272,0 | 253,0 | 207,2 | 214,4 |
| $\text{TOPK}_{\text{Wab}}$ | 285,4 | 285,4 | 274,4 | 253,2 | 207,2 | 214,4 |

Fonte: Autoria própria.

Para $\lambda = 0,25$, $\text{TOPK}_{\text{Wa}}$e $\text{TOPK}_{\text{Wab}}$apresentam queda mais acentuada ($-0,73$ e $-0,76$ p.p.), proporcionalmente. Em contrapartida, para $\lambda = 0,75$, observa-se leve vantagem de $\text{AT}_{\text{Wab}}$(+0,0648 p.p.). Interpretando esses resultados à luz da teoria ARTMAP, pode-se encontrar compatibilidade com o seguinte mecanismo:

- Em deslocamentos moderados, parte dos protótipos transferidos pode permanecer útil, reduzindo necessidade de criar novas categorias (ou acelerando a convergência supervisionada);

Tabela 10 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 0{,}25$): acurácia média (±DP) por fração de dados-alvo.

| Fração | ZS | TFS | AT–Wa | AT–WaWab | AT–TopK(WaWab) |
|---|---|---|---|---|---|
| 0,05 | 0,7647±0,0000 | 0,9444±0,0082 | 0,9448±0,0081 | 0,9446±0,0082 | 0,9448±0,0081 |
| 0,10 | 0,7647±0,0000 | 0,9434±0,0065 | 0,9431±0,0065 | 0,9434±0,0065 | 0,9431±0,0065 |
| 0,20 | 0,7647±0,0000 | 0,9442±0,0066 | 0,9440±0,0066 | 0,9442±0,0066 | 0,9440±0,0066 |
| 0,40 | 0,7647±0,0000 | 0,9448±0,0054 | 0,9447±0,0054 | 0,9448±0,0054 | 0,9447±0,0054 |
| 0,60 | 0,7647±0,0000 | 0,9436±0,0048 | 0,9436±0,0048 | 0,9436±0,0048 | 0,9436±0,0048 |
| 0,80 | 0,7647±0,0000 | 0,9476±0,0032 | 0,9476±0,0032 | 0,9476±0,0032 | 0,9476±0,0032 |
| 1,00 | 0,7647±0,0000 | 0,9477±0,0035 | 0,9477±0,0035 | 0,9477±0,0035 | 0,9401±0,0185 |

Fonte: Autoria própria.

Tabela 11 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 0{,}25$): F1 médio (±DP) por fração de dados-alvo.

| Fração | ZS | TFS | AT–Wa | AT–WaWab | AT–TopK(WaWab) |
|---|---|---|---|---|---|
| 0,05 | 0,0000±0,0000 | 0,8710±0,0252 | 0,8724±0,0249 | 0,8716±0,0252 | 0,8724±0,0249 |
| 0,10 | 0,0000±0,0000 | 0,8705±0,0197 | 0,8699±0,0198 | 0,8705±0,0197 | 0,8699±0,0198 |
| 0,20 | 0,0000±0,0000 | 0,8728±0,0209 | 0,8724±0,0210 | 0,8728±0,0209 | 0,8724±0,0210 |
| 0,40 | 0,0000±0,0000 | 0,8759±0,0182 | 0,8757±0,0182 | 0,8759±0,0182 | 0,8757±0,0182 |
| 0,60 | 0,0000±0,0000 | 0,8730±0,0160 | 0,8730±0,0160 | 0,8730±0,0160 | 0,8730±0,0160 |
| 0,80 | 0,0000±0,0000 | 0,8821±0,0103 | 0,8821±0,0103 | 0,8821±0,0103 | 0,8821±0,0103 |
| 1,00 | 0,0000±0,0000 | 0,8822±0,0114 | 0,8822±0,0114 | 0,8822±0,0114 | 0,8624±0,0510 |

Fonte: Autoria própria.

Tabela 12 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 1{,}00$): acurácia média (±DP) por fração de dados-alvo.

| Fração | ZS | TFS | AT–Wa | AT–WaWab | AT–TopK(WaWab) |
|---|---|---|---|---|---|
| 0,05 | 0,7686±0,0000 | 0,9518±0,0062 | 0,9518±0,0062 | 0,9518±0,0062 | 0,9518±0,0062 |
| 0,10 | 0,7686±0,0000 | 0,9444±0,0116 | 0,9444±0,0116 | 0,9444±0,0116 | 0,9444±0,0116 |
| 0,20 | 0,7686±0,0000 | 0,9449±0,0104 | 0,9449±0,0104 | 0,9449±0,0104 | 0,9449±0,0104 |
| 0,40 | 0,7686±0,0000 | 0,9448±0,0084 | 0,9448±0,0084 | 0,9448±0,0084 | 0,9448±0,0084 |
| 0,60 | 0,7686±0,0000 | 0,9450±0,0083 | 0,9450±0,0083 | 0,9450±0,0083 | 0,9450±0,0083 |
| 0,80 | 0,7686±0,0000 | 0,9447±0,0078 | 0,9447±0,0078 | 0,9447±0,0078 | 0,9447±0,0078 |
| 1,00 | 0,7686±0,0000 | 0,9447±0,0073 | 0,9447±0,0073 | 0,9447±0,0073 | 0,9447±0,0073 |

Fonte: Autoria própria.

- Em *shifts* específicos (como o observado em $\lambda = 0,25$), a seleção TopK pode ter privilegiado protótipos menos representativos do alvo, limitando cobertura do espaço e aumentando erros iniciais;
- Em deslocamentos maiores ($\lambda \to 1$), as vantagens da transferência praticamente desaparecem e há leve degradação em algumas variantes, coerente com o aumento de *mismatch* e a necessidade de plasticidade (criação de novas categorias) para acomodar padrões deslocados.

Tabela 13 – Curvas de aprendizagem no domínio $\mathbf{T}_2(\lambda)$ (regularização $\lambda = 1{,}00$): F1 médio (±DP) por fração de dados-alvo.

| Fração | ZS | TFS | AT–Wa | AT–WaWab | AT–TopK(WaWab) |
|---|---|---|---|---|---|
| 0,05 | 0,0000±0,0000 | 0,8906±0,0210 | 0,8906±0,0210 | 0,8906±0,0210 | 0,8906±0,0210 |
| 0,10 | 0,0000±0,0000 | 0,8724±0,0343 | 0,8724±0,0343 | 0,8724±0,0343 | 0,8724±0,0343 |
| 0,20 | 0,0000±0,0000 | 0,8734±0,0316 | 0,8734±0,0316 | 0,8734±0,0316 | 0,8734±0,0316 |
| 0,40 | 0,0000±0,0000 | 0,8734±0,0266 | 0,8734±0,0266 | 0,8734±0,0266 | 0,8734±0,0266 |
| 0,60 | 0,0000±0,0000 | 0,8738±0,0267 | 0,8738±0,0267 | 0,8738±0,0267 | 0,8738±0,0267 |
| 0,80 | 0,0000±0,0000 | 0,8723±0,0249 | 0,8723±0,0249 | 0,8723±0,0249 | 0,8723±0,0249 |
| 1,00 | 0,0000±0,0000 | 0,8722±0,0244 | 0,8722±0,0244 | 0,8722±0,0244 | 0,8722±0,0244 |

Fonte: Fonte: Autoria própria.

### 5.3.2 Complexidade vs. desempenho ao variar $\lambda$

A Tabela 9 mostra redução do número de categorias com o aumento de $\lambda$ em quase todas as estratégias de transferência (por exemplo, $\mathrm{AT}_{\mathrm{Wa}}$: 338,8 $\rightarrow$ 266,8). Esse resultado sugere que o *shift* controlado induzido altera a dinâmica de comprometimento de categorias de forma não trivial: em alguns regimes, a FAM pode consolidar menos categorias (possivelmente por maior sobreposição de regiões no espaço normalizado, ou por mudanças na distribuição que reduzem necessidade de fragmentação sob a vigilância adotada). Ainda assim, como as diferenças de acurácia não são muito relevantes, o principal uso prático dessa análise é evidenciar que a transferência pode impactar fortemente a estrutura interna (categorias) sem necessariamente se traduzir em ganhos proporcionais de desempenho.

### 5.3.3 Curvas de aprendizado no domínio $\mathbf{T}_2(\lambda)$

Em $\lambda = 0{,}25$, as curvas de TFS, AT–Wa e AT–WaWab se sobrepõem amplamente ao longo das frações, permanecendo no intervalo aproximado de $0{,}94$–$0{,}95$ em acurácia. Por exemplo, na fração $0{,}05$, TFS atinge Acc. 0,9444±0,0082 e AT–Wa Acc. 0,9448±0,0081, o que indica ausência de ganho sistemático com transferência nessa condição. Na fração $1{,}00$, TFS permanece em Acc. 0,9477±0,0035, enquanto a variante seletiva AT–TopK(WaWab) reduz para Acc. 0,9401±0,0185 com maior variabilidade, o que sugere que a seleção Top-k pode descartar categorias úteis quando há abundância de dados no domínio-alvo, apesar de ser a estratégia de transferência com melhor acurácia/ nº de categorias.

O mesmo padrão é observado para F-1: para $\lambda = 0{,}25$ e fração $1{,}00$, TFS apresenta F-1 0,8822±0,0114, enquanto AT–TopK(WaWab) tem F-1 0,8624±0,0510, com desvio-padrão significativamente maior. Portanto, no regime completo, a seletividade extrema não traz benefícios claros e pode piorar a estabilidade.

Em $\lambda = 1{,}00$, as métricas permanecem igualmente próximas entre TFS e variantes de transferência. No regime completo, por exemplo, TFS atinge Acc. 0,9447±0,0073 e F1 0,8722±0,0244. Isso reforça a interpretação de que, em graus elevados de deslocamento entre domínios (maior $\lambda$) a dinâmica de ajuste no domínio-alvo passa a ser dominada por $\lambda$, reduzindo o impacto da

inicialização via transferência sobre as métricas agregadas.

Por fim, o baseline ZS apresenta acurácia em torno de $0{,}765$–$0{,}769$ e F1 igual a zero, o que é compatível com uma decisão degenerada (por exemplo, predição constante de uma classe em um problema desbalanceado). Ainda assim, ZS fornece uma referência inferior clara: qualquer forma de aprendizado no domínio-alvo (TFS ou AT) produz ganho substancial em relação a essa linha de base.

## 5.4 TRANSFERÊNCIA SELETIVA (TOP-k E PODA)

A transferência seletiva busca reduzir o risco de transferência negativa, evitando a reutilização integral de categorias/pesos pouco pertinentes ao domínio-alvo. Foram avaliadas duas abordagens: seleção Top-$\mathbf{k}$, que retém as categorias mais relevantes segundo o critério de pureza apresentado e poda, que remove/mascara categorias abaixo de um limiar.

A Tabela 14 mostra o efeito da seleção Top-$\mathbf{k}$ com limiar de pureza fixado em $0{,}95$. Ao aumentar $\mathbf{k}$, o número médio de categorias efetivamente transferidas/ativadas ($N_{cat}$) reduz-se de aproximadamente $54$ para cerca de $46$, enquanto a acurácia permanece constante no resumo (Acc. $\approx 0{,}9514$). Esse comportamento é consistente com redundância entre categorias transferidas: um subconjunto menor já concentra a informação discriminativa útil para o domínio-alvo. Em termos práticos, o resultado é relevante por permitir reduzir armazenamento e custo computacional (menor $N_{cat}$) sem perda perceptível de desempenho.

Para a poda, os resumos indicaram manutenção de $100\%$ das categorias para os limiares avaliados. Isso é compatível com um cenário em que as categorias candidatas já atendem ao critério de pureza utilizado, tornando a poda pouco efetiva. Nessa condição, Top-$\mathbf{k}$ oferece maior controle de complexidade do que uma poda por limiar.

Tabela 14 – Resultados da transferência seletiva (Top-$\mathbf{k}$) variando o número de categorias transferidas.

| Top-$\mathbf{k}$ | $N_{cat}$ (Wa) | $N_{cat}$ (WaWab) | Acc. (Wa) | Acc. (WaWab) |
|---|---|---|---|---|
| 1 | 54,0 | 54,0 | 0,9514 | 0,9514 |
| 2 | 51,3 | 51,3 | 0,9514 | 0,9514 |
| 3 | 49,3 | 49,3 | 0,9514 | 0,9514 |
| 5 | 47,3 | 47,3 | 0,9514 | 0,9514 |
| 10 | 45,7 | 45,7 | 0,9514 | 0,9514 |

Fonte: Autoria própria.

## 5.5 SENSIBILIDADE A $\rho$ E $\beta$

Na FAM, o parâmetro de vigilância $\rho$ regula o refinamento das categorias. Valores mais altos favorecem maior especialização e, tipicamente, maior número de categorias e valores mais baixos favorecem representações mais compactas. A taxa de aprendizado $\beta$ controla a intensidade

das atualizações dos pesos, influenciando a estabilidade do aprendizado e a capacidade de adaptação ao domínio-alvo. Em cenários de mudança de domínio, a combinação $(\rho, \beta)$ parece afetar simultaneamente a complexidade do modelo (via número de categorias) e a qualidade de generalização no domínio-alvo.

Para quantificar essa sensibilidade, realizou-se busca em grade em $\rho \in \{0{,}70; 0{,}80; 0{,}85; 0{,}90; 0,95\}$ e $\beta \in \{0{,}50; 0{,}80; 1{,}00\}$, considerando frações de dados-alvo de $0{,}1$ e $1{,}0$ e níveis de severidade do *shift* $\lambda \in [0; 0{,}5; 1{,}0]$ na construção de $\mathbf{T}_2(\lambda)$.

A Tabela 15 apresenta, para cada par $(\lambda, \text{fração})$, a melhor combinação $(\rho^\star, \beta^\star)$ encontrada, comparando treinamento do zero (TFS) e duas variantes de transferência: $\text{AT}_{\text{Wa}}$(transferência de $\mathbf{W}^a$) e $\text{AT}_{\text{Wab}}$(transferência de $(W_a, W_{ab})$).

Tabela 15 – Melhores combinações de $\rho$ e $\beta$.

| $\lambda$ | Fração | Método | $\rho^\star$ | $\beta^\star$ | Acc. | F1 | $N_{cat}$ |
|---|---|---|---|---|---|---|---|
| 0,00 | 0,1 | TFS | 0,85 | 0,50 | 0,9608±0,0015 | 0,9094±0,0060 | 36,0±1,4 |
| 0,00 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,85 | 0,50 | 0,9601±0,0019 | 0,9088±0,0067 | 72,7±3,1 |
| 0,00 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,85 | 0,50 | 0,9608±0,0015 | 0,9094±0,0060 | 36,0±1,4 |
| 0,00 | 1,0 | TFS | 0,85 | 0,50 | 0,9528±0,0052 | 0,8884±0,0172 | 37,3±1,5 |
| 0,00 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,90 | 0,50 | 0,9518±0,0054 | 0,8849±0,0178 | 80,0±3,0 |
| 0,00 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,85 | 0,50 | 0,9528±0,0052 | 0,8884±0,0172 | 37,3±1,5 |
| 0,50 | 0,1 | TFS | 0,85 | 0,50 | 0,9602±0,0019 | 0,9082±0,0073 | 36,0±1,4 |
| 0,50 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,85 | 0,50 | 0,9602±0,0019 | 0,9082±0,0073 | 72,7±3,1 |
| 0,50 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,85 | 0,50 | 0,9602±0,0019 | 0,9082±0,0073 | 36,0±1,4 |
| 0,50 | 1,0 | TFS | 0,85 | 0,50 | 0,9528±0,0052 | 0,8884±0,0172 | 37,3±1,5 |
| 0,50 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,90 | 0,50 | 0,9518±0,0054 | 0,8849±0,0178 | 80,0±3,0 |
| 0,50 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,85 | 0,50 | 0,9528±0,0052 | 0,8884±0,0172 | 37,3±1,5 |
| 1,00 | 0,1 | TFS | 0,95 | 0,80 | 0,9569±0,0035 | 0,9000±0,0118 | 73,0±4,4 |
| 1,00 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,85 | 1,00 | 0,9531±0,0053 | 0,8871±0,0166 | 73,0±4,4 |
| 1,00 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,70 | 0,50 | 0,9553±0,0048 | 0,8946±0,0150 | 43,0±4,4 |
| 1,00 | 1,0 | TFS | 0,95 | 0,80 | 0,9478±0,0069 | 0,8731±0,0237 | 73,0±4,4 |
| 1,00 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,85 | 1,00 | 0,9477±0,0064 | 0,8730±0,0217 | 73,0±4,4 |
| 1,00 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,95 | 0,80 | 0,9478±0,0069 | 0,8731±0,0237 | 73,0±4,4 |

Fonte: Autoria própria.

Um ponto particularmente informativo encontrado foi no regime de poucos dados com baixo deslocamento entre domínios. Em $\lambda = 0{,}00$ e fração $0{,}1$, TFS alcança $Acc.\ 0,9608 \pm 0,0015$ com $N_{cat} = 36,0 \pm 1,4$, enquanto $\text{AT}_{\text{Wa}}$apresenta $Acc.\ 0,9601 \pm 0,0019$ mas com $N_{cat} = 72,7 \pm 3,1$, isto é, uma solução substancialmente mais complexa (maior número de categorias) sem ganho correspondente em acurácia. Esse resultado sugere que a transferência de $\mathbf{W}_a$ pode induzir sobre-particionamento do espaço de entrada no domínio-alvo quando não há mecanismo adicional de controle de complexidade.

A Tabela 16 complementa a análise ao resumir a variabilidade da acurácia dentro da grade (amplitude $\text{Acc}_{max}$–$\text{Acc}_{min}$ e DP sobre a grade). Na fração $0{,}1$ e sob deslocamento nulo ou moderado ($\lambda \in \{0{,}0; 0{,}5\}$), as amplitudes podem atingir ordem de $10^{-2}$, evidenciando que a escolha de $(\rho, \beta)$ é determinante em regimes de poucos dados. Para deslocamentos mais

severos ($\lambda = 1{,}0$), observa-se redução da amplitude em parte dos cenários, sugerindo que, sob maior discrepância entre domínios, o desempenho torna-se menos sensível a variações finas de hiperparâmetros (isto é, a incerteza passa a ser dominada pelo próprio *shift* e pela disponibilidade de dados-alvo). Além disso, para $\lambda = 1{,}0$ e fração $0{,}1$, $\text{AT}_{\text{Wab}}$atinge $Acc.\ 0,9553 \pm 0,0048$ com $N_{cat} = 43,0 \pm 4,4$, compatível com soluções mais compactas neste cenário específico, ainda que $\lambda$ não atue como penalização explícita do modelo e sim como parâmetro de construção do domínio-alvo.

Tabela 16 – Sensibilidade da acurácia à variação de $\rho$ e $\beta$.

| $\lambda$ | Fração | Método | $\text{Acc}_{min}$ | $\text{Acc}_{max}$ | Amplitude | DP (grade) |
|---|---|---|---|---|---|---|
| 0,00 | 0,1 | TFS | 0,9409 | 0,9608 | 0,0199 | 0,0073 |
| 0,00 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,9353 | 0,9601 | 0,0249 | 0,0094 |
| 0,00 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,9409 | 0,9608 | 0,0199 | 0,0073 |
| 0,00 | 1,0 | TFS | 0,9364 | 0,9528 | 0,0164 | 0,0055 |
| 0,00 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,9361 | 0,9518 | 0,0157 | 0,0051 |
| 0,00 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,9364 | 0,9528 | 0,0164 | 0,0055 |
| 0,50 | 0,1 | TFS | 0,9409 | 0,9608 | 0,0199 | 0,0073 |
| 0,50 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,9353 | 0,9602 | 0,0249 | 0,0094 |
| 0,50 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,9409 | 0,9608 | 0,0199 | 0,0073 |
| 0,50 | 1,0 | TFS | 0,9346 | 0,9528 | 0,0182 | 0,0063 |
| 0,50 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,9381 | 0,9518 | 0,0137 | 0,0043 |
| 0,50 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,9346 | 0,9528 | 0,0182 | 0,0063 |
| 1,00 | 0,1 | TFS | 0,9523 | 0,9569 | 0,0046 | 0,0015 |
| 1,00 | 0,1 | $\text{AT}_{\text{Wa}}$ | 0,9491 | 0,9553 | 0,0062 | 0,0024 |
| 1,00 | 0,1 | $\text{AT}_{\text{Wab}}$ | 0,9523 | 0,9553 | 0,0030 | 0,0011 |
| 1,00 | 1,0 | TFS | 0,9432 | 0,9478 | 0,0046 | 0,0015 |
| 1,00 | 1,0 | $\text{AT}_{\text{Wa}}$ | 0,9426 | 0,9488 | 0,0062 | 0,0024 |
| 1,00 | 1,0 | $\text{AT}_{\text{Wab}}$ | 0,9432 | 0,9478 | 0,0046 | 0,0015 |

Fonte: Autoria própria.

## 5.6 SÍNTESE DOS RESULTADOS

A partir da análise dos resultados experimentais obtidos, pode-se destacar:

1. O deslocamento medido entre os domínios aumenta com $\lambda$, (principalmente em MMD e na distância $\mathbf{T}_1$–$\mathbf{T}_2(\lambda)$), validando o desenho experimental de $\mathbf{T}_2(\lambda)$ proposto;

2. Em $\mathbf{T}_1$, o baseline TFS é altamente competitivo, e os ganhos de transferência são pequenos, sugerindo que, neste conjunto, a FAM se adapta rapidamente sem precisar de forte reutilização estrutural;

3. Transferir diretamente $\mathbf{W}^a$ tende a aumentar a complexidade (mais categorias), coerente com a teoria ART/ARTMAP: protótipos não perfeitamente alinhados ao alvo aumentam *resets* e a criação de novas categorias para manter consistência supervisionada;

4. Estratégias de redução/seleção (Top-K) controlam a proliferação de categorias e mantêm desempenho semelhante ao baseline (marginalmente superior), embora possam apresentar transferência negativa pontual quando a seleção não prioriza protótipos relevantes para um $\lambda$ específico;

5. Transferir $\mathbf{W}^{ab}$, não altera substancialmente os resultados. O efeito observado é pequeno e, quando presente, aparece apenas em cenários localizados (por exemplo, leve melhora em $\lambda = 0,75$);

6. A escolha de $(\rho, \beta)$ é particularmente crítica em regimes de poucos dados e deslocamento ($\lambda$) baixo/moderado , onde a variabilidade dentro da grade é maior;

7. O deslocamento induzido por $\lambda$ reduz a sensibilidade a hiperparâmetros e tende a estabilizar a complexidade do modelo;

8. A transferência seletiva Top-$\mathbf{k}$ é útil como mecanismo de compactação (redução de $N_{cat}$) com manutenção de acurácia em faixas de $\mathbf{k}$ moderadas, mas pode degradar desempenho e estabilidade quando aplicada de forma muito agressiva no regime de dados completo.

Esses resultados sustentam a discussão central da tese sobre transferência estrutural na FAM: o benefício não deve ser avaliado apenas por métricas finais (acurácia/$F_1$), mas também pela estrutura interna (número de categorias) e pelo risco de transferência negativa sob *shift* de domínio.

# 6 CONCLUSÃO

Neste trabalho, foram propostas diferentes metodologias para a realização do aprendizado por transferência na rede neural Fuzzy ARTMAP. As diferentes metodologias foram testadas em um problema de detecção de ocupação, onde o objetivo é detectar se um ambiente está sendo ocupado. Os testes foram realizados considerando um domínio fonte composto por dados reais, um domínio alvo também composto por dados reais (mas com características diferentes do domínio fonte) e um domínio alvo sintético criado a partir do domínio alvo real. As estratégias de transferência propostas foram: duas estratégias *baselines*, ZS (*zero-shot*) e TFS (*train from scratch*); estratégias de transferência direta TL-Wa (adaptação restrita) e TL-WaWab (adaptação ampliada); transferências seletivas por relevância de categorias, TopK-Wa e TopK-Wab e transferências seletivas por poda Pruned-Wa e Pruned-WaWab.

Para a análise de discrepância entre os domínios, foram utilizadas métricas como PAD, MMD e CORAL.

Por fim, os resultados foram analisados por meio de curvas de aprendizado e métricas clássicas de classificação (acurácia, $F_1$, ROC) e análise de categorias criadas.

Em síntese, os resultados indicam que, no cenário estudado, a transferência com a rede ARTMAP *Fuzzy* tende a produzir ganhos de desempenho modestos quando o treinamento supervisionado no alvo já é forte, entretanto, estratégias seletivas como Top-$\mathbf{K}$ mostram-se relevantes por reduzir complexidade (número de categorias) mantendo desempenho competitivo, sobretudo quando há discrepância significativa entre domínios e risco de proliferação de categorias durante a adaptação.

Com base nos resultados experimentais, serão listadas, de forma resumida, as principais conclusões na seção abaixo.

## 6.1 PRINCIPAIS CONCLUSÕES

O gerador do domínio sintético $\mathbf{T}_2(\lambda)$ foi consistente em induzir deslocamento progressivo. As métricas de discrepância entre $\mathbf{T}_1$ e $\mathbf{T}_2(\lambda)$ apresentaram tendência de aumento com $\lambda$, em especial a MMD (e, em menor escala, a CORAL), indicando que o mecanismo de construção do domínio-alvo sintético produziu um *shift* controlado, conforme o objetivo metodológico.

A discrepância entre o domínio-fonte $\mathbf{S}$ e os domínios de destinos é elevada, o que limita ganhos de transferência direta. A PAD para $\mathbf{S}$ versus $T_1/T_2(\lambda)$ permaneceu próxima do máximo teórico (valores próximos de 2), sugerindo forte separabilidade entre as distribuições após o pré-processamento adotado. Esse cenário é compatível com maior ocorrência de *mismatch* e *reset* na FAM, com potencial aumento na criação de categorias ao readaptar no alvo.

Em $\mathbf{T}_1$, a baseline *Train From Scratch* (TFS) já é muito forte; os ganhos médios de acurácia com transferência foram marginais. No regime supervisionado completo (fração $= 1{,}0$), o

TFS alcançou desempenho elevado e estável, tornando difícil obter melhorias expressivas por transferência. As variantes de transferência apresentaram, em média, ganhos pequenos frente à variabilidade do protocolo experimental, o que sugere que, para este conjunto e configuração, a FAM adapta-se eficientemente ao alvo mesmo sem reutilização explícita do modelo fonte.

A rede FAM consegue manter robustez na transferência de conhecimento para todas as estratégias testadas e sobre todos os deslocamentos de domínio ($\lambda$) testados, sem degradar ou perder o conhecimento adquirido, de maneira geral. Para tanto, na transferência para os domínios alvos, aumentou significativamente o número de categorias criadas para garantir o agrupamento de protótipos relativamente divergentes. Esses resultados reforçam a teoria base da FAM, de aprendizado incremental e estabilidade/plasticidade.

O principal efeito das estratégias de transferência em $\mathbf{T}_1$ foi na complexidade do modelo (número de categorias), e não no desempenho. As estratégias baseadas em transferência de protótipos (*Wa*/*WaWab*) tenderam a elevar o número de categorias, especialmente em frações pequenas, sem ganhos proporcionais de acurácia, evidenciando um *trade-off* desempenho–complexidade. Em contrapartida, a estratégia Top-$\mathbf{K}$ mostrou-se efetiva como mecanismo de regularização estrutural: manteve desempenho comparável ao TFS e reduziu a proliferação de categorias, preservando um modelo mais compacto.

A estratégia *zero-shot* (ZS) mostrou-se inadequada para o alvo, evidenciando forte impacto do *shift*. A avaliação direta do modelo treinado em $\mathbf{S}$ sobre o alvo, sem adaptação, apresentou degradação substancial (incluindo colapso na classe positiva refletido por $F_1$ nulo), reforçando que o desalinhamento entre domínios é suficiente para inviabilizar generalização direta.

Em $\mathbf{T}_2(\lambda)$, o aumento do *shift* altera sobretudo a relação desempenho–complexidade, com diferenças de desempenho entre estratégias tipicamente pequenas em relação ao TFS. Ao comparar com TFS no regime fração $= 1{,}0$, as variações de acurácia das estratégias de transferência permaneceram, em geral, próximas de zero (exceto ZS, consistentemente inferior). A análise por categorias indicou que o *shift* pode influenciar o custo/complexidade de adaptação: estratégias com Top-$\mathbf{K}$ e/ou mecanismos seletivos tendem a reduzir o número de categorias em relação à transferência direta.

A sensibilidade a hiperparâmetros ($\rho$ e $\beta$) foi moderada e, sob *shift* mais severo, tende a ser dominada pela discrepância de domínio e pela fração de dados do alvo. Observou-se que combinações similares de $(\rho, \beta)$ se repetem como boas escolhas em diferentes cenários, e que, para deslocamentos mais intensos, variações finas de hiperparâmetros exercem menor impacto relativo quando comparadas ao efeito do próprio *shift* e à disponibilidade de dados rotulados no domínio-alvo.

## 6.2 CONTRIBUIÇÕES DO TRABALHO

Podem ser citadas, como as principais contribuições do trabalho:

- proposição e implementação de um *pipeline* reprodutível de transferência com FAM, incluindo baselines e estratégias seletivas (Top-$\mathbf{K}$ e poda);
- construção de um domínio-alvo sintético $\mathbf{T}_2(\lambda)$ para análise controlada de severidade de *shift*;
- teste do potencial da rede ARTMAP Fuzzy em realizar transferência de conhecimento estrutural;
- análise conjunta de desempenho preditivo e complexidade (número de categorias) como critério prático para avaliar transferência de conhecimento em redes ART/ARTMAP;
- uso complementar de métricas de discrepância (PAD, MMD e CORAL) para interpretar resultados sob a ótica de desalinhamento entre domínios.

## 6.3 LIMITAÇÕES

Algumas limitações principais devem ser destacadas, primeiro, a avaliação empírica concentrou-se em dois datasets de ocupação e em uma tarefa binária (depois da transformação do domínio fonte), o que restringe a generalização das conclusões para outros domínios e configurações (por exemplo, multiclasse). Segundo, o domínio sintético $\mathbf{T}_2(\lambda)$ modela *shift* por transformações paramétricas e mistura convexa, o que pode não capturar toda a complexidade de *shift* real (por exemplo, deriva de conceito e mudanças estruturais não lineares).

## 6.4 TRABALHOS FUTUROS

Como continuidade deste estudo, sugere-se ampliar a avaliação para outros pares de domínios e tarefas (incluindo cenários multiclasse e diferentes tipos de *shift*), principalmente em cenários em que a rede FAM não consiga um desempenho elevado em TFS, sendo possível assim, separar mais adequadamente os níveis de ganho. Outra sugestão é investigar estratégias adicionais de seletividade/regularização na FAM, incluindo penalizações explícitas para controlar a criação de categorias e critérios alternativos de relevância no Top-$\mathbf{K}$.

# REFERÊNCIAS


AL-OBAYDY, W. N. I.; SUANDI, S. A. Open-set single-sample face recognition in video surveillance using fuzzy ARTMAP. **Neural Computing and Applications**, v. 32, n. 5, p. 1405–1412, 2020.

ALEIXO, E. L. *et al.* Catastrophic forgetting in deep learning: a comprehensive taxonomy. **Journal of the Brazilian Computer Society**, v. 30, n. 1, p. 175–211, 2024.

ANDONIE, R.; SASU, L. Fuzzy ARTMAP with input relevances. **IEEE Transactions on Neural Networks**, v. 17, n. 4, p. 929–941, 2006.

BALZANO, L.; NOWAK, R. Blind calibration of sensor networks. In: **Proceedings of the 6th International Conference on Information Processing in Sensor Networks (IPSN)**. [S.l.: s.n.], 2007. p. 79–88.

BAXTER, J. A model of inductive bias learning. **Journal of Artificial Intelligence Research**, v. 12, p. 149–198, 2000.

BEN-DAVID, S. *et al.* Analysis of representations for domain adaptation. In: **Advances in Neural Information Processing Systems 19**. [S.l.: s.n.], 2006. p. 137–144.

BEN-DAVID, S. *et al.* A theory of learning from different domains. **Machine Learning**, v. 79, n. 1, p. 151–175, 2010.

BISHOP, C. M. Training with noise is equivalent to Tikhonov regularization. **Neural Computation**, v. 7, n. 1, p. 108–116, 1995. Disponível em: https://direct.mit.edu/neco/article/7/1/108/5828/Training-with-Noise-is-Equivalent-to-Tikhonov. Acesso em: 5 jan. 2026.

BLITZER, J.; MCDONALD, R.; PEREIRA, F. Domain adaptation with structural correspondence learning. In: **Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Sydney, Australia: Association for Computational Linguistics, 2006. p. 120–128.

BOMMASANI, R. *et al.* **On the opportunities and risks of foundation models**. [S.l.], 2021. Disponível em: https://crfm.stanford.edu/assets/report.pdf.

BOUSMALIS, K. *et al.* Unsupervised pixel-level domain adaptation with generative adversarial networks. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2017. p. 3722–3731.

CANDANEDO, L. **Occupancy detection**. 2016. Dataset. Dataset. Disponível em: https://doi.org/10.24432/C5X01N.

CARLINI, N. *et al.* Extracting training data from large language models. In: **30th USENIX Security Symposium (USENIX Security 21)**. [s.n.], 2021. p. 2633–2650. Disponível em: https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

CARPENTER, G. A.; GROSSBERG, S. A massively parallel architecture for a self-organizing neural pattern recognition machine. **Computer Vision, Graphics, and Image Processing**, v. 37, n. 1, p. 54–115, 1987.

CARPENTER, G. A. *et al.* Fuzzy ARTMAP: a neural network architecture for incremental supervised learning of analog multidimensional maps. **IEEE Transactions on Neural Networks**, v. 3, n. 5, p. 698–713, 1992.

CARPENTER, G. A.; GROSSBERG, S.; REYNOLDS, J. H. ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. **Neural Networks**, v. 4, n. 5, p. 565–588, 1991.

CARPENTER, G. A.; GROSSBERG, S.; ROSEN, D. B. Fuzzy ART: fast stable learning and categorization of analog patterns by an adaptive resonance system. **Neural Networks**, v. 4, n. 6, p. 759–771, 1991.

CARUANA, R. Multitask learning. **Machine Learning**, v. 28, n. 1, p. 41–75, 1997.

CHEN, T. *et al.* A simple framework for contrastive learning of visual representations. In: **Proceedings of the 37th International Conference on Machine Learning (ICML)**. [S.l.]: PMLR, 2020.

CONNOLLY, J.-F.; GRANGER, É.; SABOURIN, R. Supervised incremental learning with the fuzzy ARTMAP neural network. In: PREVOST, L.; MARINAI, S.; SCHWENKER, F. (Ed.). **Artificial Neural Networks in Pattern Recognition (ANNPR 2008)**. Berlin, Heidelberg: Springer, 2008, (Lecture Notes in Computer Science, v. 5064). p. 66–77.

DAI, W. *et al.* Boosting for transfer learning. In: **Proceedings of the 24th International Conference on Machine Learning (ICML)**. [S.l.: s.n.], 2007. p. 193–200.

DAY, O.; KHOSHGOFTAAR, T. M. A survey on heterogeneous transfer learning. **Journal of Big Data**, v. 4, p. 29, 2017.

DEVLIN, J. *et al.* BERT: pre-training of deep bidirectional transformers for language understanding. In: **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)**. [S.l.: s.n.], 2019.

DONAHUE, J. *et al.* DeCAF: a deep convolutional activation feature for generic visual recognition. In: **Proceedings of the 31st International Conference on Machine Learning (ICML)**. [S.l.]: PMLR, 2014. p. 647–655.

DOSOVITSKIY, A. *et al.* An image is worth 16x16 words: transformers for image recognition at scale. In: **International Conference on Learning Representations (ICLR)**. [S.l.: s.n.], 2021.

ERICSSON, L. *et al.* How well do self-supervised models transfer? In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2021.

FINN, C.; ABBEEL, P.; LEVINE, S. Model-agnostic meta-learning for fast adaptation of deep networks. In: **Proceedings of the 34th International Conference on Machine Learning (ICML)**. [S.l.]: PMLR, 2017.

GANIN, Y. *et al.* Domain-adversarial training of neural networks. **Journal of Machine Learning Research**, v. 17, n. 59, p. 1–35, 2016.

GOKHALE, T.; FEINGLASS, J.; YANG, Y. Covariate shift detection via domain interpolation sensitivity. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [s.n.], 2022. Disponível em: https://openreview.net/forum?id=YkPjTHZDdm.

GOODFELLOW, I. *et al.* Generative adversarial nets. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2014.

GRETTON, A. *et al.* A kernel method for the two-sample problem. In: **Advances in Neural Information Processing Systems 19**. [S.l.: s.n.], 2007. p. 513–520.

GRETTON, A. *et al.* A kernel two-sample test. **Journal of Machine Learning Research**, v. 13, p. 723–773, 2012.

GROSSBERG, S. How does a brain build a cognitive code? **Psychological Review**, v. 87, n. 1, p. 1–51, 1980.

GUAN, H.; LIU, M. Domain adaptation for medical image analysis: a survey. **IEEE Transactions on Biomedical Engineering**, v. 69, n. 3, p. 1173–1185, 2022.

GULRAJANI, I.; LOPEZ-PAZ, D. In search of lost domain generalization. In: **International Conference on Learning Representations (ICLR)**. [s.n.], 2021. Disponível em: https://openreview.net/forum?id=lQdXeXDoWtI.

HE, K. *et al.* Momentum contrast for unsupervised visual representation learning. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2020.

HE, K. *et al.* Deep residual learning for image recognition. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2016. p. 770–778.

HOULSBY, N. *et al.* Parameter-efficient transfer learning for NLP. In: **Proceedings of the 36th International Conference on Machine Learning (ICML)**. [S.l.: s.n.], 2019.

HOWARD, J.; RUDER, S. Universal language model fine-tuning for text classification. In: **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)**. [S.l.: s.n.], 2018.

HU, E. J. *et al.* LoRA: low-rank adaptation of large language models. In: **International Conference on Learning Representations (ICLR)**. [S.l.: s.n.], 2022.

HU, W. *et al.* Strategies for pre-training graph neural networks. In: **International Conference on Learning Representations (ICLR)**. [S.l.: s.n.], 2020.

III, H. D. Frustratingly easy domain adaptation. In: **Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)**. [S.l.: s.n.], 2007. p. 256–263.

JUDD, C. H. The relation of special training to general intelligence. **Educational Review**, v. 36, p. 28–42, 1908.

KINGMA, D. P.; WELLING, M. Auto-encoding variational bayes. In: **International Conference on Learning Representations (ICLR)**. [s.n.], 2014. Disponível em: https://openreview.net/forum?id=33X9fd2-9FyZd.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. ImageNet classification with deep convolutional neural networks. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2012.

LONG, M. *et al.* Learning transferable features with deep adaptation networks. In: **Proceedings of the 32nd International Conference on Machine Learning (ICML)**. [S.l.]: PMLR, 2015.

LONG, M. *et al.* Conditional adversarial domain adaptation. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2018.

MITCHELL, T. M. **The need for biases in learning generalizations**. [S.l.], 1980.

NIU, S. *et al.* A decade survey of transfer learning (2010–2020). **IEEE Transactions on Artificial Intelligence**, v. 1, n. 2, p. 151–166, 2020.

NUNES, R. F. **Classificação de imagens baseada no aprendizado profundo e na teoria da ressonância adaptativa**. 2024. Tese (Tese (Doutorado em Engenharia Elétrica)) — Universidade Estadual Paulista (Unesp), Faculdade de Engenharia, Ilha Solteira, Ilha Solteira, SP, Brasil, 2024.

OQUAB, M. *et al.* Learning and transferring mid-level image representations using convolutional neural networks. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2014. p. 1717–1724.

PAN, S. J. *et al.* Domain adaptation via transfer component analysis. **IEEE Transactions on Neural Networks**, v. 22, n. 2, p. 199–210, 2011.

PAN, S. J.; YANG, Q. A survey on transfer learning. **IEEE Transactions on Knowledge and Data Engineering**, v. 22, n. 10, p. 1345–1359, 2010.

PERKINS, D. N.; SALOMON, G. **Transfer of learning**. Oxford, U.K.: Pergamon, 1992.

PRATT, L. Y. Discriminability-based transfer between neural networks. In: **Advances in Neural Information Processing Systems 5 (NIPS 1992)**. [S.l.: s.n.], 1992. p. 204–211.

QUIÑONERO-CANDELA, J. *et al.* (Ed.). **Dataset shift in machine learning**. Cambridge, MA: MIT Press, 2009.

RAINA, R. *et al.* Self-taught learning: transfer learning from unlabeled data. In: **Proceedings of the 24th International Conference on Machine Learning (ICML)**. [S.l.: s.n.], 2007. p. 759–766.

ROSENFELD, E.; RAVIKUMAR, P.; RISTESKI, A. An online learning approach to interpolation and extrapolation in domain generalization. In: **Proceedings of The 25th International Conference on Artificial Intelligence and Statistics (AISTATS)**. [S.l.]: PMLR, 2022. v. 151, p. 2641–2657.

RUSU, A. A. *et al.* **Progressive neural networks**. 2016. ArXiv preprint arXiv:1606.04671.

SCARSELLI, F. *et al.* The graph neural network model. **IEEE Transactions on Neural Networks**, v. 20, n. 1, p. 61–80, 2009.

SCHAT, E. *et al.* The data representativeness criterion: predicting the performance of supervised classification based on data set similarity. **PLOS ONE**, v. 15, n. 8, p. e0237009, 2020.

SILVA, L. E. B. da; WUNSCH, D. C. A survey of adaptive resonance theory neural network models for engineering applications. **Neural Networks**, v. 120, p. 167–203, 2019.

SINGH, A.; CHAUDHARI, S. **Room occupancy estimation**. 2018. Dataset. Dataset. Disponível em: https://doi.org/10.24432/C5P605.

SNELL, J.; SWERSKY, K.; ZEMEL, R. Prototypical networks for few-shot learning. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2017.

STRUBELL, E.; GANESH, A.; MCCALLUM, A. Energy and policy considerations for deep learning in NLP. In: **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)**. [s.n.], 2019. Disponível em: https://aclanthology.org/P19-1355/.

SUGIYAMA, M.; KAWANABE, M. **Machine learning in non-stationary environments: introduction to covariate shift adaptation**. Cambridge, MA: MIT Press, 2012.

SUN, B.; FENG, J.; SAENKO, K. Correlation alignment for unsupervised domain adaptation. In: **Domain Adaptation in Computer Vision Applications**. [S.l.]: Springer, 2017. p. 153–171.

SUN, B.; SAENKO, K. Deep CORAL: correlation alignment for deep domain adaptation. In: **Computer Vision – ECCV 2016 Workshops**. [S.l.]: Springer, 2016. p. 443–450.

TAYLOR, M. E.; STONE, P. Transfer learning for reinforcement learning: a survey. **Journal of Machine Learning Research**, v. 10, p. 1633–1685, 2009.

THRUN, S.; MITCHELL, T. M. Lifelong robot learning. **Robotics and Autonomous Systems**, v. 15, n. 1–2, p. 25–46, 1995.

THRUN, S.; PRATT, L. (Ed.). **Learning to learn**. New York, NY: Springer, 1998.

TZENG, E. *et al.* Adversarial discriminative domain adaptation. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2017. p. 2962–2971.

VASWANI, A. *et al.* Attention is all you need. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2017.

VEDANTAM, R.; LOPEZ-PAZ, D.; SCHWAB, D. J. An empirical investigation of domain generalization with empirical risk minimizers. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2021.

VINYALS, O. *et al.* Matching networks for one shot learning. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2016.

WANG, L. *et al.* Parameter-efficient fine-tuning in large language models: a survey of methodologies. **Artificial Intelligence Review**, v. 58, p. 227, 2025.

WANG, L. *et al.* A comprehensive survey of continual learning: theory, method and application. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 46, n. 8, p. 5362–5383, 2024.

WANG, R. *et al.* Trans-LoRA: towards transferable parameter-efficient fine-tuning. In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2024.

WANG, Y. *et al.* Blind drift calibration of sensor networks using signal space projection and Kalman filter. In: **Proceedings of the 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)**. [S.l.: s.n.], 2015. p. 1–6.

WANG, Z. *et al.* Characterizing and avoiding negative transfer. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2019. p. 11293–11302.

WEENINK, D. Category ART: a variation on adaptive resonance theory neural networks. In: IFA. **IFA Proceedings**. Amsterdam: IEEE, 1997. v. 21, p. 117–129.

WEISS, K.; KHOSHGOFTAAR, T. M.; WANG, D. A survey of transfer learning. **Journal of Big Data**, v. 3, p. 9, 2016.

YOSINSKI, J. *et al.* How transferable are features in deep neural networks? In: **Advances in Neural Information Processing Systems (NeurIPS)**. [S.l.: s.n.], 2014.

YU, X. *et al.* Transfer learning for medical images analyses: a survey. **Neurocomputing**, v. 489, p. 230–254, 2022.

ZHANG, H. *et al.* Mixup: beyond empirical risk minimization. In: **International Conference on Learning Representations (ICLR)**. [s.n.], 2018. Disponível em: https://arxiv.org/abs/1710.09412.

ZHANG, W. *et al.* A survey on negative transfer. **IEEE/CAA Journal of Automatica Sinica**, v. 10, n. 2, p. 305–329, 2023.

ZHAO, S. *et al.* A comparison review of transfer learning and self-supervised learning: definitions, applications, advantages and limitations. **Expert Systems with Applications**, v. 242, p. 122807, 2024.

ZHOU, D.-W. *et al.* Continual learning with pre-trained models: a survey. In: **Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)**. [S.l.: s.n.], 2024. p. 8363–8371.

ZHU, J.-Y. *et al.* Unpaired image-to-image translation using cycle-consistent adversarial networks. In: **Proceedings of the IEEE International Conference on Computer Vision (ICCV)**. [S.l.: s.n.], 2017.

ZHUANG, F. *et al.* A comprehensive survey on transfer learning. **Proceedings of the IEEE**, v. 109, n. 1, p. 43–76, 2021.

# DADOS CURRICULARES

| IDENTIFICAÇÃO | |
|---|---|
| | Danieli Biagi Vilela<br>17/05/1994. |
| **Nacionalidade** | Brasileira |
| **Nome em citações bibliográficas** | Vilela, Danieli<br>Vilela, D. |
| **Currículo Lattes** | http://lattes.cnpq.br/0532010012226064 |
| **ORCID** | https://orcid.org/0000-0003-3501-9644 |
| **FORMAÇÃO ACADÊMICA** | |
| **2013/2017** | Graduação em Engenharia de Produção<br>UFMS |
| **2018/2021** | Mestrado em Engenharia Elétrica<br>UNESP - FEIS |
| **PRODUÇÃO BIBLIOGRÁFICA** | |
| BIAGI VILELA, DANIELI; PLASENCIA LOTUFO, ANNA DIVA. Título: Detecção de Presença a Partir de Dados de Sensores Utilizando Rede Neural ARTMAP Fuzzy. *In:* Congresso Brasileiro de Automática, 2020. **Anais** [. . . ] | |
| **PARTICIPAÇÃO EM BANCAS E ORIENTAÇÕES** | |
| **Bancas de trabalhos de conclusão** | |
| LOPES, M. L. M.; VILELA, D. B.; FERREIRA, A. B. A. Participação em banca de Pedro Augusto do Amaral Brigido. **DESENVOLVIMENTO DE MODELO PARA PREDIÇÃO DE FALHAS EM MÁQUINAS E EQUIPAMENTOS INDUSTRIAIS USANDO ALGORITMOS DE MACHINE LEARNING**. 2023. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho, Ilha Solteira, 2023. | |
| LOTUFO, A. D.; BIAGI, D.V.; FORMAGGIO, G. P. Participação em banca de Renan Ferreira Felix. **PREVISÃO DE PREÇOS DE COMERCIALIZAÇÃO DE ENERGIA ELÉTRICA POR REDES NEURAIS ARTIFICIAIS**. 2022. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho, Ilha Solteira, 2022. | |
| LOTUFO, A. D.; BIAGI, D.V.; FORMAGGIO, G. P. Participação em banca de Luiz Francisco Feltrim Junior. **PREVISÃO DE PREÇOS DE AÇÕES DA BOLSA DE VALORES POR REDES NEURAIS ARTIFICIAIS**. 2022. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) - Universidade Estadual Paulista Júlio de Mesquita Filho, Ilha Solteira, 2022. | |
| **PARTICIPAÇÃO EM EVENTOS CIENTÍFICOS** | |
| Congresso Brasileiro de Automática, XXIV, 2022, (*Online*). Detecção de Presença a Partir de Dados de Sensores Utilizando Rede Neural ARTMAP Fuzzy. Congresso Brasileiro de Automática. 2022 (Seminário). | |