UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE - CCENS
COLEGIADO DO CURSO DE CIÊNCIA DA COMPUTAÇÃO

Christian Prenholato da Rocha Lima

União de dois modelos de Redes Neurais Convolucionais utilizando técnicas de extração de modelos

Alegre, ES

2025

Christian Prenholato da Rocha Lima

União de dois modelos de Redes Neurais Convolucionais utilizando técnicas de extração de modelos

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Ciência da Computação.

Universidade Federal do Espírito Santo – UFES

Centro de Ciências Exatas, Naturais e da Saúde

Colegiado do Curso de Ciência da Computação

Orientador: Prof. D. Sc. Jacson Rodrigues Correia da Silva

Alegre, ES

2025

Christian Prenholato da Rocha Lima

União de dois modelos de Redes Neurais Convolucionais utilizando técnicas de extração de modelos

Trabalho de Conclusão de Curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em 22 de agosto de 2025.

COMISSÃO EXAMINADORA

Jacson R C Silva

Prof. Dr. Jacson Rodrigues Correia da Silva
Universidade Federal do Espírito Santo
Orientador

Edmar Hell Kampke

Prof. Dr. Edmar Hell Kampke
Universidade Federal do Espírito Santo

JP Pirovani

Prof.ª Dra. Juliana Pinheiro Campos Pirovani
Universidade Federal do Espírito Santo

Resumo

Modelos de aprendizado profundo têm oferecido soluções eficazes em diferentes domínios, impulsionando avanços relevantes em aplicações que impactam positivamente a sociedade. Com a crescente adoção desses modelos, muitas empresas passaram a disponibilizá-los como serviços por meio de APIs. No entanto, estudos demonstram que esses modelos apresentam vulnerabilidades e podem ter seu conhecimento extraído, possibilitando a criação de modelos substitutos com desempenho semelhante aos modelos originais. Um exemplo é o ataque *Copycat*, que permite gerar um modelo substituto a partir de imagens naturais fora do domínio do problema, rotuladas via API do modelo-alvo, facilitando o processo de extração. Diante desse cenário, este trabalho investiga a hipótese de que é possível construir um modelo substituto mais robusto por meio da extração do conhecimento de dois modelos-alvo e da união desse conhecimento em um único modelo, de forma a aumentar sua acurácia e capacidade de generalização para diferentes domínios. Para isso, foram realizados experimentos utilizando dois grupos de conjuntos de dados: (i) *MNIST* e *SVHN*, e (ii) *Office-Home* (*Art* e *Real*). Foram desenvolvidos cinco modelos *copycats* para cada grupo de conjuntos de dados, aplicando diferentes técnicas de combinação de conhecimento extraído utilizando abordagens do *continual learning*, como *knowledge distillation*, *model ensemble* e *replay-based*, utilizando a concatenação em ordem aleatória do treinamento. Os resultados mostram que, em cada grupo de domínios, a técnica mais eficaz para combinar os conhecimentos variou. No grupo *MNIST/SVHN*, o modelo denominado *copycat-kd-from-copycats* alcançou a melhor acurácia média (83,16%) nos *datasets* de teste. Já no domínio *Office-Home*, o melhor desempenho foi obtido pelo modelo *copycat-combined*, com acurácia média de 78,44%. Tais resultados comprovam a hipótese do trabalho, demonstrando que é possível unir o conhecimento de dois modelos extraídos, atingindo até 95% do desempenho de um modelo treinado diretamente com os dados de treinamento de ambos os domínios.

Palavras-chaves: Redes Neurais Convolucionais, Destilação de Conhecimento, Extração de Modelos, União de Modelos, Adaptação de Domínio, Aprendizado Contínuo.

Abstract

Deep learning models have provided effective solutions across different domains, driving significant advances in applications that positively impact society. With the increasing adoption of these models, many companies have begun offering them as services through APIs. However, studies show that these models are vulnerable and their knowledge can be extracted, enabling the creation of substitute models with performance similar to that of the original models. One example is the Copycat attack, which allows generating a substitute model from natural images outside the problem domain, labeled via the target model's API, facilitating the extraction process. In this context, this work investigates the hypothesis that it is possible to construct a more robust substitute model by extracting knowledge from two target models and combining it into a single model, thereby improving its accuracy and generalization across different domains. To this end, experiments were conducted using two groups of datasets: (i) MNIST and SVHN, and (ii) Office-Home (Art and Real). Five copycat models were developed for each dataset group, applying different techniques to combine the extracted knowledge using continual learning approaches such as knowledge distillation, model ensemble, and replay-based strategies, with randomized concatenation during training. The results show that, for each domain group, the most effective technique to combine knowledge varied. In the MNIST/SVHN group, the model named copycat-kd-from-copycats achieved the highest average accuracy (83.16%) on the test datasets. In the Office-Home domain, the best performance was obtained by the copycat-combined model, with an average accuracy of 78.44%. These results confirm the work's hypothesis, demonstrating that it is possible to merge the knowledge of two extracted models, reaching up to 95% of the performance of a model trained directly with the training data from both domains.

Keywords: Convolutional Neural Networks, Model Extraction, Knowledge distillation, Union of Models, Domain Adaptation, Continual Learning.

Lista de ilustrações

Figura 1 – Representação do Perceptron 17
Figura 2 – Exemplo de Rede Neural 17
Figura 3 – Exemplo de imagens do ImageNet 22
Figura 4 – Exemplo de imagens do MNIST 23
Figura 5 – Exemplo de imagens do SVHN 24
Figura 6 – Exemplo de imagens do Office-Home 25
Figura 7 – Representação das redes BAN 26
Figura 8 – Ataque Copycat. 28
Figura 9 – Geração do cenário de ataque 34
Figura 10 – Fluxo das abordagens de continual learning 35

Lista de tabelas

Tabela 1 – Detalhes sobre o treinamento dos oráculos: número de imagens utilizadas no treinamento e teste dos modelos oráculos. 33
Tabela 2 – Acurácia dos modelos nos datasets de teste do MNIST e SVHN e a média entre eles 39
Tabela 3 – Acurácia relativa dos modelos copycats em relação a maior acurácia dos modelos oráculos no dominio do MNIST e SVHN 40
Tabela 4 – Acurácia dos Modelos oráculos e copycats nos datasets de teste de cada dominio e suas médias no dominio Office-Home Art e Real 40
Tabela 5 – Acurácia relativa dos modelos copycats em relação a acurácia dos modelos oracle-real, oracle-art e oracle-combined 41

Lista de abreviaturas e siglas

MNIST	Modified National Institute of Standards and Technology database
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
SVHN	Street View House Numbers
API	Application Programming Interface
INA	Imagens Naturais Aleatórias
RR	Rótulos Roubados
IA	Inteligência Artificial
RNA	Rede Neural Artificial
RNC	Rede Neural Convolucional
ML	Machine Learning
CL	Continual Learning
KD	Knowledge Distillation
BAN	Born Again Networks

Sumário

1 Introdução **9**
1.1 Motivação e Justificativa 12
1.2 Objetivos 13
1.2.1 Objetivo geral 13
1.2.2 Objetivos específicos 13
2 Referencial Teórico **15**
2.1 Aprendizado de Máquina 15
2.2 Redes Neurais 16
2.3 Redes Neurais Convolucionais 18
2.4 Datasets 20
2.5 Transferência de Aprendizado 24
2.6 Extração de Modelos 27
2.7 União de Modelos e Continual Learning 29
3 Metodologia **31**
3.1 Conjuntos de dados 31
3.2 Modelos oráculos 32
3.3 Extração de conhecimento com Copycat CNN 33
3.4 Treinamento dos modelos Copycats 34
3.5 Avaliação dos modelos 36
3.6 Configuração de *Hardware* e *Software* 36
4 Resultados **38**
4.1 Extração de modelos 38
4.1.1 Domínios MNIST e SVHN 38
4.1.2 Domínios Office-Home Art e RealWorld 40
4.2 Discussão 42
5 Conclusão **44**

Referências **46**

1 Introdução

O campo do *Deep Learning* tem proporcionado avanços significativos em diversas áreas, oferecendo soluções inovadoras que beneficiam a sociedade. Entre as técnicas mais proeminentes estão as Redes Neurais Convolucionais (RNCs), amplamente utilizadas em tarefas de classificação e reconhecimento de padrões em imagens (LORENTE; RIERA; RANA, 2021).

Para contextualizar o presente estudo, algumas definições fundamentais são apresentadas. RNCs (*Redes Neurais Convolucionais*) são redes neurais projetadas para processar dados estruturados em grades, como imagens, extraindo automaticamente características hierárquicas por meio de operações de convolução. O termo modelo refere-se a qualquer rede neural treinada capaz de realizar previsões ou classificações a partir de dados de entrada. Extração de modelo é o processo pelo qual um modelo substituto aprende a replicar o comportamento de um modelo-alvo, geralmente utilizando suas respostas como referência. O oráculo é o modelo-alvo original, tratado como uma *black-box*, cujo conhecimento se deseja capturar. Finalmente, o termo *Copycat* descreve o modelo substituto obtido através da extração, que busca imitar o desempenho do oráculo.

No setor de saúde, as RNCs já foram aplicadas no diagnóstico de doenças por meio da análise de imagens médicas, auxiliando profissionais na identificação precoce de condições como câncer e doenças neurológicas. Por exemplo, um estudo recente utilizou RNCs para prever a idade cerebral a partir de imagens de ressonância magnética, contribuindo para a detecção de processos neurodegenerativos em estágios iniciais (OLIVEIRA et al., 2021).

Na área de gestão de desastres naturais, as RNCs foram aplicadas para a predição de enchentes, utilizando dados históricos e séries temporais. Essas redes neurais artificiais analisam informações, como o nível máximo de rios, e convertem esses dados em representações bidimensionais, permitindo o uso de redes convolucionais e recorrentes. Essa abordagem possibilita a identificação de padrões e a previsão de eventos críticos, contribuindo para a implementação de estratégias preventivas, aumento da segurança da população em áreas de risco e mitigação dos impactos econômicos e sociais causados por enchentes (MESQUITA; UEYAMA, 2024). As RNCs também têm sido aplicadas na área industrial, realizando a detecção de não conformidades em equipamentos, o que contribui para a manutenção preditiva e a redução de falhas operacionais (JUNIOR; CAURIN, 2022).

As RNCs foram inicialmente introduzidas por LeCun et al. (1998) para o reconhecimento de dígitos manuscritos. No entanto, sua adoção em larga escala ocorreu apenas recentemente, impulsionada pelo avanço do hardware e, consequentemente , pela evolução

das arquiteturas (SRINIVAS et al., 2016). Um marco nesse progresso foi a AlexNet, que revolucionou a visão computacional ao vencer a competição *ImageNet Large Scale Visual Recognition Challenge (ILSVRC)* em 2012 (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Esse modelo demonstrou o impacto do uso de redes mais profundas no desempenho de tarefas de classificação de imagens, influenciando diretamente o desenvolvimento de arquiteturas como VGG e ResNet, que ampliaram a capacidade das RNCs de extrair características relevantes e superar abordagens tradicionais (SIMONYAN; ZISSERMAN, 2015; HE et al., 2015).

Além da evolução das arquiteturas, outro fator essencial para o avanço das RNCs foi a disponibilidade de grandes conjuntos de dados. A *AlexNet*, por exemplo, obteve um desempenho muito acima dos trabalhos da época ao ser treinada no *ImageNet*, um dos maiores e mais relevantes *datasets* para aprendizado profundo, contendo milhões de imagens rotuladas em milhares de categorias (RUSSAKOVSKY et al., 2015). A existência de bases de dados extensas e diversificadas permitiu que modelos mais complexos fossem treinados de maneira eficaz, promovendo avanços significativos na generalização das redes neurais. A qualidade e diversidade dos *datasets* utilizados no treinamento das RNCs influenciam diretamente seu desempenho. Alguns conjuntos amplamente adotados, como *ImageNet* (RUSSAKOVSKY et al., 2015), *Modified National Institute of Standards and Technology database (MNIST)* (LECUN et al., 1998) e *Street View House Numbers (SVHN)* (NETZER et al., 2011), fornecem bases sólidas para a construção de modelos robustos, tornando sua curadoria um fator crítico no avanço da visão computacional.

Dado seu avanço na área, é comum encontrar modelos de aprendizado de máquina disponíveis na internet, refletindo o significativo progresso da área. Esses modelos são conhecidos como *white-box*, pois permitem que os usuários façam o download de uma representação completa para uso local (TRAMèR et al., 2016). Isso significa que tanto a arquitetura do modelo quanto seus parâmetros estão acessíveis publicamente, possibilitando sua análise, modificação e reutilização em diferentes contextos.

Modelos *white-box* são interessantes porque permitem a modificação e reutilização para finalidades diferentes daquelas para as quais foram originalmente desenvolvidos. Para isso, uma técnica amplamente utilizada é o *transfer learning*, que consiste em transferir o conhecimento adquirido em um domínio para melhorar o desempenho de um modelo em um domínio relacionado (WEISS; KHOSHGOFTAAR; WANG, 2016). Essa técnica, quando aplicada com sucesso, melhora o desempenho do aprendizado ao evitar esforços dispendiosos de rotulagem de dados, permitindo o reaproveitamento de modelos pré-treinados em novas tarefas, o que reduz o consumo de recursos computacionais e acelera o desenvolvimento (PAN; YANG, 2009).

Para ilustrar a viabilidade do *transfer learning*, Weiss, Khoshgoftaar e Wang (2016) apresentam uma analogia baseada em experiências do mundo real. Duas pessoas querem

aprender a tocar piano: uma sem nenhuma experiência musical prévia e outra com amplo conhecimento musical por já tocar violão. A pessoa com formação musical será capaz de aprender piano de forma mais eficiente, transferindo o conhecimento musical previamente adquirido para essa nova tarefa. Da mesma forma, um modelo de aprendizado de máquina é capaz de aproveitar informações de uma tarefa previamente aprendida e usá-las de forma benéfica para aprender uma tarefa relacionada (WEISS; KHOSHGOFTAAR; WANG, 2016).

Os modelos de *black-box*, diferentemente dos modelos de *white-box*, são acessíveis apenas por meio de interfaces de consulta, sem expor informações sobre parâmetros internos, arquitetura ou dados de treinamento (TRAMèR et al., 2016). Com a popularização da Inteligência Artificial (IA) como produto comercial, tornou-se prática comum disponibilizar modelos exclusivamente via *Application Programming Interfaces (APIs)*, permitindo que usuários enviem consultas e recebam previsões sem acesso direto ao funcionamento interno. Essa configuração, utilizada em aplicações como reconhecimento de imagens, tradução automática e assistentes virtuais, busca proteger a propriedade intelectual das empresas e garantir maior controle sobre a utilização do sistema (CORREIA-SILVA, 2023).

Um exemplo de modelo *black-box* amplamente difundido é o ChatGPT (OPENAI, 2023), acessível apenas via API ou interface de uso fornecida pela OpenAI, sem disponibilização de seus parâmetros internos. Já o LLaMA (TOUVRON et al., 2023), desenvolvido pela Meta, é um exemplo de modelo *white-box*, pois possui pesos e arquitetura abertamente disponibilizados para a comunidade de pesquisa.

Apesar dessas medidas, esse formato apresenta vulnerabilidades. Adversários podem explorar a interface de consulta para realizar ataques de extração, replicando o comportamento do modelo original e treinando um substituto com desempenho similar, mesmo sem acesso aos dados de treinamento (TRAMèR et al., 2016; OREKONDY; SCHIELE; FRITZ, 2018; CORREIA-SILVA et al., 2018). Estudos mostram que RNCs, amplamente aplicadas nesses cenários, são especialmente suscetíveis à extração e à replicação de conhecimento.

Mais preocupante ainda é a possibilidade de ataques em larga escala, em que diferentes modelos, extraídos de múltiplas organizações, possam ser combinados em um único sistema. Nesse caso, o adversário não apenas reproduz funcionalidades já existentes, mas gera um novo modelo potencialmente mais robusto e competitivo, colocando em risco tanto a vantagem comercial quanto a segurança dos detentores originais. Esse contexto evidencia a importância de compreender não apenas os mecanismos de extração, mas também como o conhecimento extraído pode ser integrado entre diferentes domínios, mesmo na ausência dos dados originais de treinamento.

Diante desse cenário, torna-se fundamental analisar as vulnerabilidades inerentes aos modelos de *black-box*, de modo a viabilizar o desenvolvimento de estratégias que não só dificultem a extração, mas também impeçam que informações provenientes de múltiplas

fontes sejam combinadas por agentes adversários. A correta compreensão dessas questões contribuem para a formulação de mecanismos de defesa mais eficazes, garantindo maior proteção à confidencialidade e à integridade dos modelos de IA.

Em muitos casos, o objetivo não é simplesmente replicar um modelo existente, mas sim adaptá-lo a um novo domínio ou incorporar novas amostras dentro do mesmo contexto. No entanto, esse processo é desafiador devido ao fenômeno conhecido como *catastrophic forgetting* (KIRKPATRICK et al., 2017), no qual o modelo, ao ser treinado para novas tarefas, perde drasticamente o desempenho nas tarefas previamente aprendidas (WANG et al., 2024).

Por este motivo, existem abordagens como o *Continual Learning (CL)* que são técnicas que buscam permitir que modelos aprendam de forma incremental, incorporando novos conhecimentos sem comprometer aqueles previamente adquiridos. Para isso, diferentes estratégias são adotadas, como a manipulação da estrutura do modelo, o uso de algoritmos especializados e técnicas que controlam a adaptação do aprendizado ao longo do tempo Rusu et al. (2022), Wang et al. (2024), Kirkpatrick et al. (2017), Frikha et al. (2022), Dietterich (2000).

Idealmente, um sistema inteligente deveria ser capaz de aprender de forma contínua, preservando o conhecimento adquirido anteriormente. No entanto, a indisponibilidade dos *datasets* originais torna esse processo ainda mais desafiador, dificultando a adaptação a novos domínios e aumentando o risco de degradação do desempenho em tarefas previamente aprendidas.

Uma alternativa para mitigar esse problema é a geração de novos dados de treinamento para viabilizar o treinamento de um novo modelo. Nesse contexto, as técnicas de extração de modelos desempenham um papel fundamental, pois permitem a obtenção do conhecimento de um modelo-alvo por meio de imagens aleatórias, possibilitando a criação de um modelo substituto que preserve as características do original sem a necessidade de acesso direto aos dados de treinamento originais (CORREIA-SILVA et al., 2018; OREKONDY; SCHIELE; FRITZ, 2018). Assim, a hipótese deste trabalho é que:

Utilizando técnicas de extração de modelos e de aprendizado contínuo, é possível gerar um modelo substituto capaz de incorporar o conhecimento de dois modelos black-box de mesmo domínio, mas com características internas distintas.

1.1 Motivação e Justificativa

O crescente uso de RNCs em aplicações comerciais tem impulsionado investimentos significativos em inteligência artificial, que somaram aproximadamente 1,067 trilhões de dólares nos últimos cinco anos (MASLEJ et al., 2024). Esse cenário evidencia tanto o valor

estratégico desses modelos quanto a necessidade de protegê-los contra ameaças emergentes, sobretudo em relação à propriedade intelectual. O desenvolvimento de modelos de IA envolve altos custos e a criação de *datasets* exclusivos, que representam ativos críticos para empresas que buscam garantir sua vantagem competitiva (CORREIA-SILVA, 2023). Entretanto, tais modelos permanecem vulneráveis a ataques de extração, nos quais agentes mal-intencionados podem replicá-los sem acesso direto aos dados originais (TRAMèR et al., 2016; OREKONDY; SCHIELE; FRITZ, 2018).

A gravidade do problema se intensifica quando se considera a possibilidade de replicação e integração de múltiplos modelos extraídos. Nesse cenário, adversários poderiam reunir o conhecimento de diferentes soluções proprietárias em um único modelo, explorando dados e recursos de terceiros sem incorrer em custos de desenvolvimento. Mais do que uma simples cópia, a união de modelos pode resultar em sistemas com desempenho ou versatilidade superiores aos originais, ampliando o risco de prejuízos competitivos e ameaçando diretamente a sustentabilidade econômica das organizações.

Diante desse contexto, o estudo da extração e integração de modelos mostra-se relevante não apenas pela contribuição acadêmica, mas também pelo impacto prático em termos de segurança e inovação tecnológica. Compreender se é possível consolidar múltiplos modelos — inclusive no cenário *black-box*, como discutido por Orekondy, Schiele e Fritz (2018) e Correia-Silva et al. (2018) — é essencial para avaliar os riscos associados e desenvolver técnicas que mitiguem tais ameaças. Além disso, essa investigação possibilita avanços na adaptação contínua de modelos sem a necessidade dos dados originais, estabelecendo bases sólidas para o desenvolvimento de soluções robustas, escaláveis e seguras.

1.2 Objetivos

Nas subseções seguintes são apresentados o objetivo geral e os objetivos específicos deste trabalho.

1.2.1 Objetivo geral

Utilizar técnicas de extração de modelos e CL para treinar um modelo de RNC capaz de inferir respostas a partir de dois modelos extraídos de domínios semelhantes, mas com características distintas, avaliando sua performance em relação aos modelos oráculos.

1.2.2 Objetivos específicos

1. Selecionar dois *datasets* da literatura que compartilhem similaridades em termos de domínio, além de um terceiro dataset fora do domínio do problema.

2. Treinar dois modelos de RNCs utilizando os *datasets* selecionados, além de um terceiro modelo que combine os dados dos dois conjuntos anteriores.

3. Aplicar uma técnica de extração, submetendo um conjunto de imagens de um *dataset* externo aos modelos oráculos e registrando suas respectivas predições.

4. Treinar modelos de RNCs utilizando os *datasets* de ataque rotulados com as predições dos modelos oráculos, incorporando a técnicas de CL para adaptação entre os diferentes domínios.

5. Avaliar o desempenho dos modelos copycats em comparação com os modelos oráculos utilizando como métrica a acurácia do modelo em cada domínio e a média entre eles.

2 Referencial Teórico

Nas seções seguintes são apresentados os principais conceitos, fundamentos e trabalhos relacionados que embasam o desenvolvimento desta pesquisa. Inicialmente, serão apresentados os princípios de aprendizado de máquina, seguidos pelos conceitos fundamentais das redes neurais. Em seguida, aprofunda-se nas Redes Neurais Convolucionais (RNCs), prosseguindo com a descrição dos *datasets* utilizados. Posteriormente, aborda-se a transferência de aprendizado, a extração de modelos e, por fim, a integração entre modelos e técnicas de *continual learning*.

2.1 Aprendizado de Máquina

O aprendizado de máquina (*Machine Learning (ML)*) é uma subárea da IA que visa o desenvolvimento de algoritmos capazes de aprender padrões e tomar decisões com base em dados, sem a necessidade de instruções explícitas para cada tarefa (SAMUEL, 1959). Essa abordagem tem se tornado uma ferramenta essencial em diversas áreas, como reconhecimento de fala, visão computacional, processamento de linguagem natural e análise de grandes volumes de dados (MITCHELL, 1997).

Dado um conjunto de dados, os algoritmos de ML utilizam essas informações para criar modelos capazes de realizar previsões sobre novas observações, isto é, dado um novo exemplo, o modelo pode inferir um resultado com base no conhecimento extraído dos dados anteriores (MONARD; BARANAUSKAS, 2003). Esses algoritmos podem ser classificados em três categorias principais: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço (RUSSELL; NORVIG, 2021).

No aprendizado supervisionado, o algoritmo é treinado com um conjunto de exemplos previamente rotulados, permitindo que o modelo, após o treinamento, classifique corretamente novos exemplos não rotulados (MONARD; BARANAUSKAS, 2003). Embora em muitos casos a obtenção dos rótulos exija a intervenção humana, o termo "supervisionado" continua sendo aplicado mesmo quando os rótulos são gerados automaticamente (GOODFELLOW; BENGIO; COURVILLE, 2016).

No aprendizado não supervisionado, o algoritmo analisa os dados fornecidos e busca identificar características semelhantes para agrupar os dados, formando clusters que representam as estruturas de semelhanças entre eles (DIKE et al., 2018). Goodfellow, Bengio e Courville (2016) descreve que a distinção entre algoritmos supervisionados e não supervisionados não é rigidamente definida, pois não existe um teste objetivo para distinguir se um valor é um recurso ou um alvo fornecido por um supervisor. Informalmente,

o aprendizado não supervisionado envolve tentativas de extrair informações de uma distribuição sem a necessidade de anotações humanas. Esse termo é frequentemente associado a tarefas como estimativa de densidade, aprendizado para extrair amostras de uma distribuição, redução de ruído nos dados ou agrupamento dos dados em categorias relacionadas (GOODFELLOW; BENGIO; COURVILLE, 2016).

Já no aprendizado por reforço, o sistema recebe recompensas quando toma decisões corretas e penalizações quando erra, buscando otimizar seu comportamento por meio de tentativa e erro (COPPIN, 2004). Esse método se baseia na atribuição de crédito e culpa, permitindo que o algoritmo aprenda as ações mais adequadas para maximizar sua recompensa ao longo do tempo.

Esses métodos de aprendizado são fundamentais para o desenvolvimento de RNCs, que utilizam algoritmos de ML para otimizar seu desempenho em diversas tarefas. A próxima seção se dedica a explicar o funcionamento das redes neurais convolucionais e como elas são treinadas no contexto do aprendizado supervisionado.

2.2 Redes Neurais

O elemento fundamental de uma rede neural é o neurônio, inspirado nas redes biológicas, onde a ativação de um neurônio resulta no envio de sinais para outros neurônios interconectados (ZHOU; LIU, 2021). Analogamente, um neurônio artificial processa entradas numéricas, combinando-as com pesos e aplicando uma função de ativação para produzir uma saída. Um exemplo clássico é o *perceptron*, que recebe entradas x_n ponderadas por pesos w_n, realiza o somatório $\sum_{i=0}^{n} w_i x_i$ e aplica uma função degrau para gerar uma saída binária, representando duas classes distintas (Figura 1). O treinamento do perceptron consiste em ajustar iterativamente os pesos sempre que uma classificação incorreta ocorrer, até que todas as instâncias do conjunto de treinamento sejam classificadas (MITCHELL, 1997).

Embora eficiente para problemas linearmente separáveis, o perceptron não consegue resolver tarefas não lineares, como a função XOR. Para lidar com padrões mais complexos, desenvolvem-se redes neurais multicamadas (*multilayer perceptrons*), nas quais neurônios são organizados em camadas de entrada, ocultas e saída (Figura 2). Cada neurônio calcula uma função $y_i = f(\sum_i w_i x_i + b_j)$, onde b_j é o viés que permite ajustar a saída da camada.

O treinamento de redes multicamadas requer métodos eficientes de ajuste de pesos. A regra Δ, baseada em gradiente descendente, atualiza os pesos proporcionalmente ao erro entre a saída prevista e o valor real, permitindo a minimização da função de perda. O algoritmo de *backpropagation* aplica essa regra de forma iterativa, propagando o erro da saída para as camadas anteriores e ajustando os pesos até a convergência (ZHOU; LIU, 2021). O gradiente descendente calcula a derivada da função de perda em relação aos

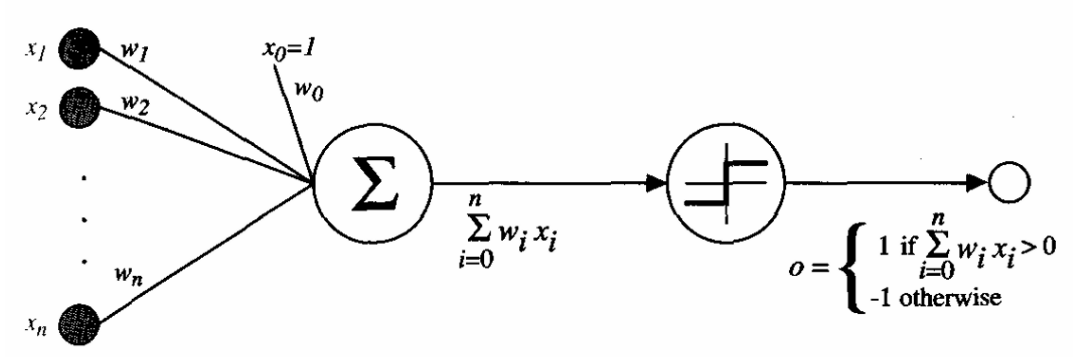

Figura 1 – O perceptron recebe entradas x_n com pesos w_n, realiza o somatório $\sum_{i=0}^{n} w_i x_i$ e aplica a função degrau (*step function*), produzindo saída $o = 1$ se $\sum w_i x_i > 0$ e -1 caso contrário (MITCHELL, 1997). Imagem do livro Machine Learning, página 87 <https://www.cs.cmu.edu/~tom/files/MachineLearningTomMitchell.pdf> (MITCHELL, 1997)

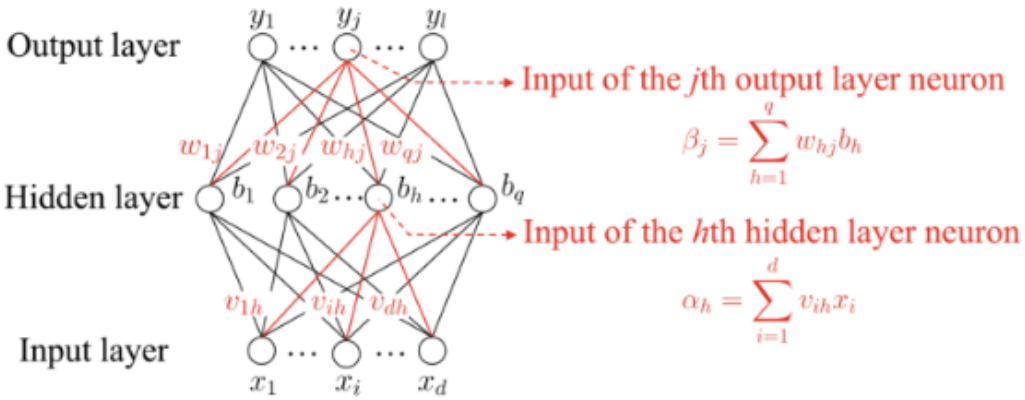

Figura 2 – Exemplo de Rede Neural. Representando a camada de entrada (*input layer*) representada pelos x_i neurônios, a camada oculta (*hidden layer*) demonstrando a entrada do neurônio h, correspondendo a $\alpha_h = \sum_{i=1}^{d} v_{ih} x_i$. E a camada de saída (*output layer*), onde o neurônio j possui a seguinte entrada, $\beta_j = \sum_{h=1}^{q} w_{hj} b_h$. Imagem do livro *Machine Learning* (ZHOU; LIU, 2021). Página 109 <https://books.google.com.br/boks?id=ctM-EAAAQBAJ>

pesos, determinando a direção e magnitude das atualizações, e diferentes funções de perda podem ser utilizadas conforme a tarefa, como entropia cruzada para classificação ou erro quadrático médio para regressão.

A avaliação do desempenho das redes neurais é feita por métricas que quantificam a capacidade de generalização do modelo. A acurácia, que mede a proporção de predições corretas, é amplamente utilizada, enquanto a taxa de erro representa a proporção de predições incorretas. Para tarefas que exigem estimativas contínuas de confiança, outras métricas, como a probabilidade logarítmica média, são mais apropriadas (GOODFELLOW; BENGIO; COURVILLE, 2016). Escolher a métrica adequada é fundamental para interpretar corretamente o desempenho das redes em diferentes tipos de problemas.

Entre as diferentes arquiteturas de redes neurais desenvolvidas para enfrentar desafios específicos, destaca-se um modelo que ganhou relevância especialmente em aplicações envolvendo imagens e sinais: as RNCs (LECUN et al., 1998). Diferentemente das redes neurais tradicionais, que operam essencialmente por meio de multiplicações de matrizes em suas camadas, as RNCs foram projetadas para explorar dados com estrutura em grade, como séries temporais unidimensionais (1D) e imagens bidimensionais (2D) (GOODFELLOW; BENGIO; COURVILLE, 2016). Sua principal característica é o uso de operações de convolução nas primeiras camadas, o que possibilita a detecção de padrões locais, sejam eles espaciais ou temporais. Essa propriedade confere às RNCs grande eficiência em tarefas como reconhecimento de imagens, explicando sua ampla adoção em cenários práticos e de pesquisa.

2.3 Redes Neurais Convolucionais

No campo do aprendizado de máquina, especialmente em tarefas de visão computacional, as RNCs têm se destacado como a arquitetura mais eficaz devido à sua capacidade de extrair hierarquias de características visuais diretamente das imagens (SZEGEDY et al., 2015). Ao longo dos anos, vários modelos de RNCs foram desenvolvidos, cada um contribuindo com avanços notáveis em estrutura, capacidade de generalização e eficiência computacional. Estes modelos, como descrito por Zhang et al. (2023), são definidos como "maquinaria computacional para ingestão de dados de um tipo, e cuspir previsões de um tipo possivelmente diferente." Isso ressalta que cada modelo de RNC é projetado para processar dados específicos e gerar previsões de acordo com as exigências de diferentes tarefas de aprendizado.

Essas redes neurais incluem pelo menos uma camada convolucional, responsável por extrair informações abstratas das imagens. Tipicamente, elas seguem uma estrutura que combina camadas convolucionais empilhadas para fornecer níveis progressivos de abstração do conteúdo da imagem original. As camadas convolucionais são frequentemente acompanhadas por operações de normalização de contraste e *pooling*, como exemplificado no modelo clássico LeNet-5 (LECUN et al., 1989). O princípio fundamental das RNCs é a aplicação de filtros (ou *kernels*) a pequenos blocos da imagem de entrada, permitindo que a rede aprenda características locais, como bordas e texturas. À medida que a rede avança para camadas mais profundas, essas características locais são combinadas para capturar informações mais complexas, como formas e objetos completos.

Os principais componentes que organizam essas redes incluem:

- Camadas convolucionais: Responsáveis por aplicar filtros na imagem de entrada para detectar características visuais. A profundidade dessas camadas define a hierarquia das características capturadas. Modelos como o AlexNet introduziram a ideia de

múltiplas camadas convolucionais empilhadas para capturar características cada vez mais abstratas. Como mencionado por Goodfellow et al. (2014) que percebeu que o desempenho dessa abordagem aumenta com a profundidade da rede convolucional

- *Pooling*: Frequentemente usado após as camadas convolucionais para reduzir a dimensionalidade dos mapas de características, reduzindo a complexidade computacional ao mesmo tempo que tenta manter as informações mais relevantes. Segundo Goodfellow, Bengio e Courville (2016), uma função de *pooling* substitui a saída da rede em um determinado local por uma estatística resumida das saídas próximas, reduzindo a dimensionalidade dos dados sem perder informações essenciais. Entre outras funções de *pooling*, existe o *max-pooling* por exemplo, que a operação seleciona a saída máxima dentro de uma vizinhança retangular, similar ao filtro da camada convolucional.

- Camadas totalmente conectadas (*Fully Connected Layers - FC layers*) são amplamente utilizadas em redes neurais profundas. Nessas camadas, cada neurônio é conectado a todos os neurônios da camada anterior, o que resulta em muitos parâmetros aprendíveis (ZHANG et al., 2023). Geralmente, essas camadas são posicionadas após as camadas convolucionais e de *pooling*, e sua função é combinar as características extraídas das camadas anteriores para realizar a classificação ou predição final.

- As funções de ativação determinam se um neurônio deve ser ativado, calculando a soma ponderada das entradas e adicionando o *bias*. Eles são operadores diferenciáveis para transformar sinais de entrada em saídas, enquanto a maioria deles adiciona não linearidade (ZHANG et al., 2023). Alguns exemplo de Funções são, ReLU (*Rectified Linear Unit*) que introduz não linearidades nas redes, permitindo que as RNCs capturem padrões complexos. Esta ativação foi um dos diferenciais do AlexNet, ajudando a acelerar o treinamento e melhorar o desempenho (KRIZHEVSKY; SUTSKEVER; HINTON, 2017).

Cada um dos componentes mencionados desempenha um papel fundamental no desempenho e na capacidade de generalização das RNCs. Um exemplo de modelo que marcou avanço significativo foi a Alexnet. Proposta por Krizhevsky, Sutskever e Hinton (2017), a AlexNet marcou um avanço significativo ao vencer o ILSVRC 2012. Sua inovação principal foi o uso de camadas convolucionais profundas e GPUs para treinamento, além de técnicas como a função de ativação ReLU e *dropout*, que ajudaram a melhorar o desempenho e reduzir o *overfitting*.

À medida que as arquiteturas se tornaram mais profundas e complexas, a necessidade de organizar camadas convolucionais de maneira mais eficiente levou ao desenvolvimento de "blocos" ou "*building blocks*", que agrupam camadas convolucionais, funções de ativação

não lineares, *pooling* e camadas totalmente conectadas em unidades reutilizáveis (ZHANG et al., 2023). Um exemplo importante dessa abordagem foi o modelo VGG, proposto por Simonyan e Zisserman (2015), que introduziu a ideia de blocos VGG. Um bloco VGG consiste em uma sequência de camadas convolucionais seguidas por uma camada de *pooling* máxima para realizar a redução da resolução espacial (ZHANG et al., 2023). Embora o custo computacional da VGG fosse elevado, sua simplicidade estrutural, aliada à profundidade, demonstrou que redes mais profundas podem melhorar substancialmente o desempenho em tarefas de classificação de imagens.

A Inception, também conhecida como GoogLeNet, foi apresentada por Szegedy et al. (2014) como uma solução para os problemas de eficiência computacional e o aumento excessivo do número de parâmetros em redes neurais profundas. A inovação principal dessa arquitetura foi a utilização de múltiplas convoluções de diferentes tamanhos (1x1, 3x3 e 5x5) em paralelo dentro de um único bloco, permitindo que a rede capturasse representações em diversas escalas ao mesmo tempo. Além disso, a introdução das convoluções de 1x1 desempenhou um papel essencial ao reduzir a dimensionalidade dos dados antes das convoluções maiores, o que, por sua vez, diminuiu significativamente o número de parâmetros, tornando o modelo mais eficiente e leve (SZEGEDY et al., 2014). Graças a essa abordagem, a Inception possibilitou a construção de redes mais profundas sem comprometer a eficiência computacional, estabelecendo um marco na evolução de arquiteturas convolucionais mais escaláveis e eficazes.

Por sua vez, a ResNet, desenvolvida por He et al. (2015), introduziu blocos residuais para mitigar problemas de degradação de desempenho em redes muito profundas. O uso desses blocos permitiu que o modelo aprendesse ajustes nos mapeamentos de identidade, facilitando a propagação do gradiente por meio de várias camadas, o que mitiga o problema do *exploding* ou *vanishing gradient*. Esse avanço possibilitou a construção de redes extremamente profundas, como a ResNet-50 e ResNet-101, sem a perda de eficiência observada em arquiteturas anteriores quando as redes aumentavam em profundidade.

Para que as redes neurais artificiais alcancem um desempenho eficiente em tarefas de aprendizado supervisionado, elas dependem da disponibilidade de dados rotulados para realizar o treinamento, chamados *datasets*. Em visão computacional, por exemplo, os datasets consistem geralmente em grandes coleções de imagens, cada uma associada a um rótulo que indica a classe à qual a imagem pertence.

2.4 Datasets

Os *datasets* são fundamentais para o treinamento e a validação de modelos de aprendizado de máquina supervisionado, pois fornecem exemplos rotulados que permitem ao modelo identificar padrões e realizar classificações (GOODFELLOW; BENGIO; COUR-

VILLE, 2016). Cada exemplo é representado por um vetor de atributos e um rótulo de classe, sendo esses dados utilizados pelo algoritmo para prever a classe de novos exemplos não rotulados (MONARD; BARANAUSKAS, 2003). A qualidade e a diversidade dos *datasets* influenciam diretamente o desempenho e a generalização dos modelos treinados, tornando essencial a escolha adequada do conjunto de dados para cada aplicação. Além disso, a disponibilidade de grandes volumes de dados anotados tem impulsionado avanços significativos na área, permitindo o desenvolvimento de modelos mais robustos e eficazes.

Geralmente, são divididos em conjuntos de treinamento, validação e teste: o primeiro ajusta os parâmetros do modelo, o segundo auxilia na escolha de hiperparâmetros e o último avalia o desempenho em dados não vistos (GOODFELLOW; BENGIO; COURVILLE, 2016). A seguir, apresentam-se alguns dos datasets mais influentes utilizados nesse contexto.

Entre os principais datasets utilizados, destaca-se o ImageNet, que se tornou um marco para o avanço da visão computacional desde seu lançamento em 2009, conforme ilustrado na Figura 3.

O ImageNet é um dos datasets mais influentes no campo da visão computacional e aprendizado de máquina, com mais de 14 milhões de imagens rotuladas distribuídas em mais de 20.000 categorias. Essa enorme quantidade de dados e diversidade de classes desempenhou um papel crucial na evolução dos modelos de reconhecimento de imagens. O marco do ImageNet é particularmente visível nas competições do ILSVRC (RUSSAKOVSKY et al., 2015), um evento anual que desafia pesquisadores e desenvolvedores a criarem modelos de aprendizado profundo capazes de classificar imagens em categorias predefinidas.

O ILSVRC foi lançado em 2010 e logo se tornou uma plataforma fundamental para medir o desempenho de modelos RNCs. No entanto, foi em 2012 que o desafio alcançou um marco histórico com a vitória da rede neural AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2017). Revolucionando o campo ao reduzir significativamente a taxa de erro na classificação de imagens, superando as abordagens anteriores em termos de precisão.

Em resumo, o ImageNet e o ILSVRC desempenharam papéis fundamentais na evolução da visão computacional, oferecendo um banco de dados diversificado e de grande escala que impulsionou a inovação e ajudou a consolidar o uso de redes neurais profundas como a técnica de escolha para tarefas de reconhecimento visual.

O *dataset* MNIST, embora seja considerado simples, desempenha um papel fundamental em experimentos e no desenvolvimento de novos algoritmos de aprendizado de máquina. Criado em 1998, ele é composto por 60.000 imagens para treinamento e 10.000 para teste, contendo dígitos manuscritos de 0 a 9 representados em matrizes de 28x28 pixels (LECUN et al., 1998). Alguns exemplos podem ser observados na Figura 4. Apesar de sua simplicidade em relação a outros conjuntos de dados, o MNIST permanece

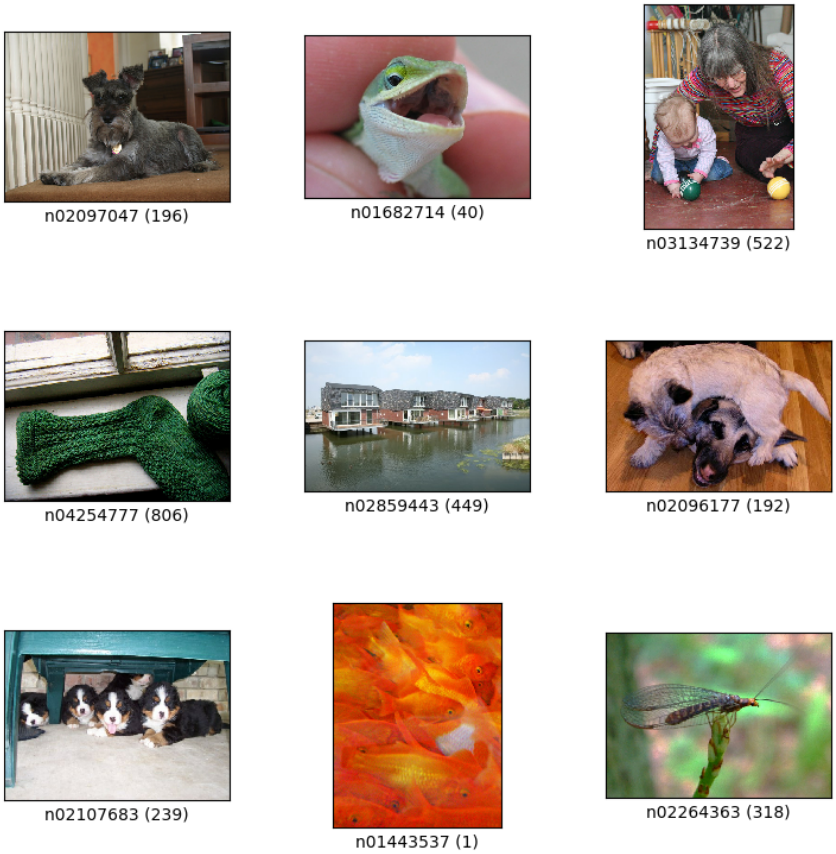

Figura 3 – Exemplo de imagens do ImageNet e seu respectivo rótulo que identifica a classe a qual pertence. Imagem de <https://www.tensorflow.org/datasets/catalog/imagenet2012?hl=pt-br>

amplamente utilizado como referência para prototipagem de modelos e estabelecimento de *benchmarks*.

Para cenários mais desafiadores, o *dataset* SVHN, disponibilizado em 2011, reúne imagens de números de residências capturadas pelo Google Street View (NETZER et al., 2011). Alguns exemplos podem ser observados na Figura 5. Com mais de 600.000 imagens rotuladas, o SVHN impõe maior complexidade em relação ao MNIST, exigindo que os modelos sejam capazes de lidar com variações significativas de iluminação, perspectiva e contexto visual.

Finalmente, o *Office-Home* (VENKATESWARA et al., 2017) é um conjunto de dados amplamente utilizado como *benchmark* para algoritmos de adaptação de domínios em tarefas de reconhecimento de objetos com aprendizado profundo. Seu principal diferencial é a representação de imagens da mesma classe em quatro domínios distintos, o que permite

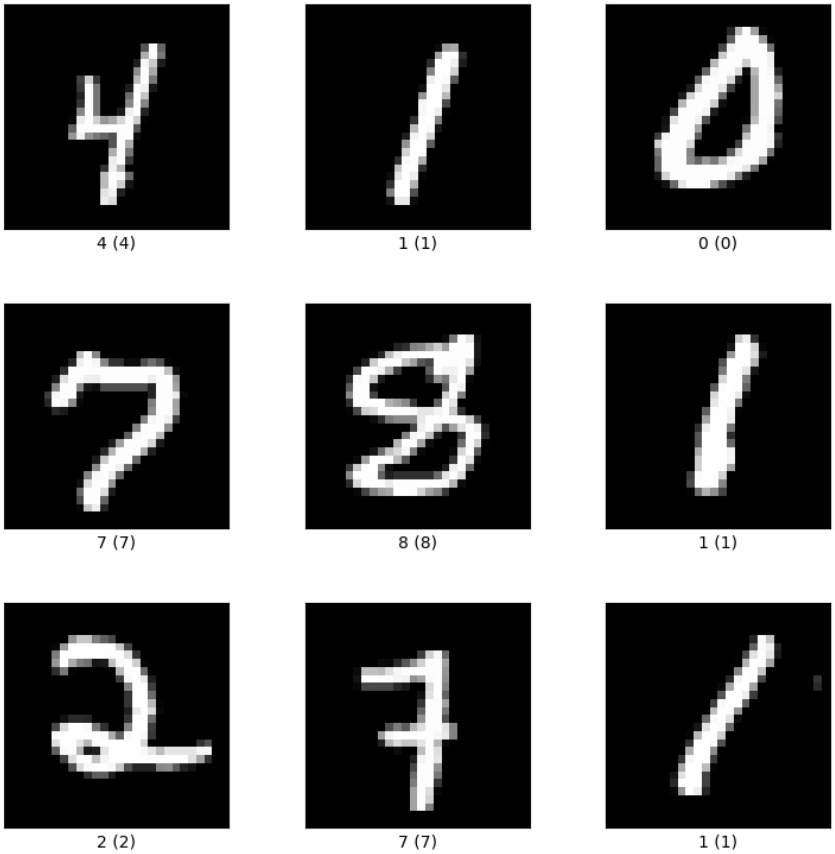

Figura 4 – Exemplo de imagens do MNIST e suas respectivas classes. Imagem de <https://www.tensorflow.org/datasets/catalog/MNIST?hl=pt-br>

avaliar a capacidade dos modelos em generalizar o aprendizado entre diferentes contextos visuais. Os quatro domínios são representados na Figura 6:

Art, que inclui imagens artísticas como esboços, pinturas e ornamentações; *Clipart*, composto por uma coleção de imagens no estilo *clipart*; *Product*, que reúne imagens de objetos sem fundo, geralmente utilizadas em catálogos de produtos; e *Real-World*, formado por fotografias de objetos capturadas com câmeras convencionais. O *dataset* é composto por 15.500 imagens, distribuídas em 65 categorias de objetos comuns em ambientes de escritório e domésticos. Essa diversidade torna o *Office-Home* uma ferramenta valiosa para o desenvolvimento e avaliação de modelos de aprendizado profundo, especialmente em cenários onde a adaptação entre domínios é crucial (VENKATESWARA et al., 2017).

Os *datasets* são essenciais para o desenvolvimento de modelos de aprendizado de máquina, pois fornecem os dados necessários para o treinamento e a validação dos modelos, permitindo a identificação de padrões e a realização de classificações (GOODFELLOW;

Figura 5 – Exemplo de imagens do SVHN cortados com um número de interesse. Imagem de <http://ufldl.stanford.edu/housenumbers>

BENGIO; COURVILLE, 2016). No entanto, a disponibilidade de grandes volumes de dados rotulados nem sempre é garantida, o que pode representar uma limitação significativa em diversos cenários. Para superar essa limitação, técnicas como a transferência de aprendizado (*transfer learning*) têm sido utilizadas para aproveitar o conhecimento adquirido por modelos pré-existentes. A transferência de aprendizado será detalhada na próxima seção, onde discutiremos suas vantagens e aplicações em maior profundidade.

2.5 Transferência de Aprendizado

A transferência de aprendizado (*Transfer Learning*) é uma técnica amplamente utilizada para reutilizar o conhecimento adquirido por uma Rede Neural Artificial (RNA) em um problema inicial e aplicá-lo em um novo domínio, reduzindo custos computacionais e a necessidade de grandes conjuntos de dados (YOSINSKI et al., 2014; PAN; YANG, 2009). Em vez de treinar um modelo do zero, é possível ajustar (*fine-tune*) uma rede

Figura 6 – Exemplo de imagens do Office-Home. Representando as classes cama, bicileta, garrafa, cadeira e óculos em cada domínio. Imagem de (VENKATESWARA et al., 2017)

pré-treinada para desempenhar bem em uma nova tarefa, preservando características úteis aprendidas anteriormente (GOODFELLOW; BENGIO; COURVILLE, 2016).

Um estudo conduzido por Yosinski et al. (2014) investigou a transferência de características em RNCs profundas, realizando um experimento que avaliou a eficácia da reutilização de camadas previamente treinadas. Primeiro, eles treinaram uma rede com oito camadas de pesos em um subconjunto das 1.000 categorias do ImageNet. Em seguida, uma nova rede da mesma arquitetura foi inicializada com as k primeiras camadas da rede pré-treinada, enquanto as camadas superiores foram inicializadas aleatoriamente. Essa nova rede foi treinada em um conjunto diferente de categorias do ImageNet, mas com menos exemplos de treinamento.

Os resultados demonstraram que as camadas iniciais das redes são responsáveis por aprender padrões genéricos, aplicáveis a diferentes tarefas, enquanto as camadas superiores especializam-se em aspectos mais específicos. Como consequência, a reutilização das camadas superiores sem ajustes adequados tende a ser menos eficaz. Além disso, foi observado que mesmo características transferidas de tarefas distantes superam inicializações aleatórias, e que a reutilização de camadas pré-treinadas melhora a generalização do modelo

após o fine-tuning (YOSINSKI et al., 2014).

O Knowledge Distillation (KD) surge como uma poderosa técnica para transferência de conhecimento entre modelos de redes neurais, destacando-se por sua capacidade de reduzir a complexidade do modelo sem comprometer o desempenho. O KD transfere informações de um modelo maior e mais complexo, conhecido como *teacher model*, para um modelo menor e mais eficiente, denominado *student model* (HINTON; VINYALS; DEAN, 2015). Este processo ocorre através do aprendizado das saídas suavizadas do modelo professor, chamadas de *soft labels*, o que permite ao modelo aluno capturar nuances e padrões que vão além da simples associação com rótulos corretos.

O KD também pode ser implementado de forma iterativa, ampliando seu potencial. Após a convergência do *teacher model*, um novo *student model* pode ser treinado não apenas para prever os rótulos corretos, mas também para replicar a distribuição de saída do professor. Este processo iterativo resulta nas chamadas Redes *Born Again Networks (BAN)*, que replicam o desempenho dos modelos professores, apresentando menores taxas de erro em tarefas como visão computacional e modelagem de linguagem (FURLANELLO et al., 2018). A Figura 7 representa o procedimento de treinamento das BANs.

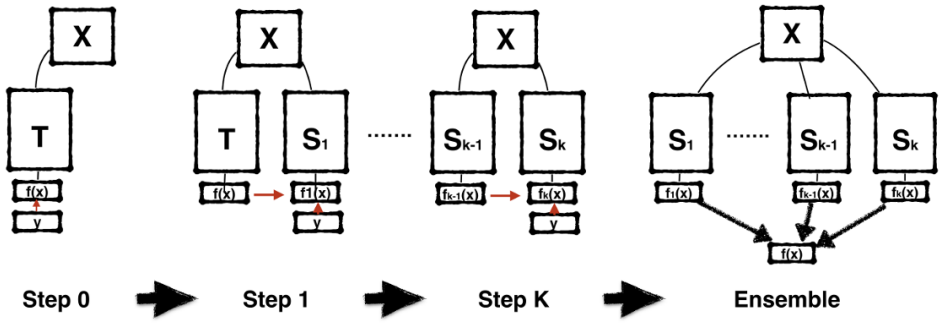

Figura 7 – Representação das redes BAN: durante o primeiro passo, o modelo de professor T é treinado a partir dos rótulos Y. Então, a cada passo consecutivo, um novo modelo idêntico é inicializado a partir de uma semente aleatória diferente e treinado a partir da supervisão da geração anterior. No final do procedimento, ganhos adicionais podem ser alcançados com um conjunto de múltiplas gerações de alunos. Imagem retirada de (FURLANELLO et al., 2018)

Após a inicialização de múltiplas gerações de modelos alunos, Furlanello et al. (2018) demonstra que a união dessas gerações pode aprimorar significativamente o desempenho dos modelos. Esse método tem sido amplamente utilizado para compactação de modelos e para cenários em que a extração de conhecimento de múltiplos modelos é necessária para criar uma única rede mais eficiente (TAN et al., 2018).

Apesar das vantagens do fine-tuning, sua aplicação depende diretamente da disponibilidade de dados representativos do novo domínio, o que pode ser um obstáculo significativo. Em contextos onde o acesso aos dados originais é limitado ou inviável, técnicas

de extração de modelos oferecem uma abordagem alternativa para capturar e transferir conhecimento sem depender dos conjuntos de dados utilizados no treinamento inicial. A próxima seção discutirá essas estratégias em detalhes.

2.6 Extração de Modelos

A extração de modelos é uma técnica que visa replicar o comportamento de um modelo de aprendizado de máquina sem ter acesso direto aos seus dados de treinamento ou parâmetros internos. Essa abordagem é particularmente útil quando o modelo original é acessível apenas por meio de uma interface de predição (*black-box*), permitindo que apenas as entradas e saídas sejam consultadas. Em contraste, modelos *white-box* são aqueles em que se tem acesso completo às informações internas, como estrutura de camadas e parâmetros, possibilitando sua utilização e modificação local sem restrições (TRAMèR et al., 2016). Além disso, a literatura também cita os modelos *gray-box*, nos quais o adversário tem acesso a informações adicionais sobre o modelo. No entanto, a maioria dos trabalhos utiliza apenas as nomenclaturas *black-box* e *white-box* (CORREIA-SILVA, 2023).

Essa técnica é especialmente valiosa em contextos onde o acesso ao dataset original é inviável, seja por custos elevados ou pela sensibilidade dos dados. Modelos de ML podem ser considerados confidenciais devido aos dados sensíveis, valor comercial ou aplicações em segurança. Muitas vezes, esses modelos são disponibilizados por meio de interfaces de consulta públicas (APIs), como em sistemas de ML como serviço ("análise preditiva"), que permitem o treinamento com dados sensíveis e cobram pelo acesso via pagamento por consulta (OREKONDY; SCHIELE; FRITZ, 2018).

Embora a extração de modelos seja útil em cenários legítimos, como o treinamento de novos modelos de IA com base no conhecimento incorporado em modelos existentes, ela também pode ser explorada de forma maliciosa. Ataques de extração permitem que agentes mal-intencionados reproduzam o comportamento de modelos oferecidos como serviço, criando modelos substitutos sem arcar com os custos de desenvolvimento (CORREIA-SILVA et al., 2018).

Uma das estratégias de extração é a *Query-Based Model Extraction* (OREKONDY; SCHIELE; FRITZ, 2018), que envolve realizar múltiplas consultas ao modelo-alvo, capturando as respostas geradas e associando-as às entradas correspondentes. Essas respostas são então usadas para treinar um novo modelo, que tenta imitar o comportamento do original. Quanto maior o número de consultas realizadas, maior a precisão da extração, permitindo que o modelo copiado atinja desempenho similar ao modelo original.

Uma variação dessa técnica é o método *Copycat CNN*, proposto por Correia-Silva (2023), que permite a extração de modelos RNCs sem a necessidade de acessar imagens do domínio específico. Em vez disso, utiliza Imagens Naturais Aleatórias (INA) para compor

o conjunto de dados de ataque. Essas imagens são passadas pela API do modelo-alvo, denominado Oráculo, que as rotula e gera os chamados "Rótulos Roubados (RR)". O modelo substituto, ou modelo *Copycat*, é então treinado com essas imagens e rótulos. Conforme ilustrado na Figura 8

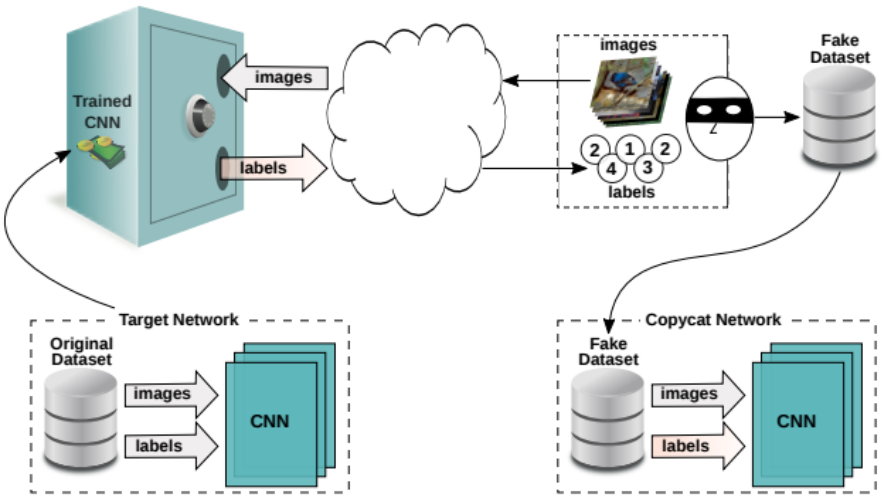

Figura 8 – À esquerda, temos o modelo-alvo, que foi treinado utilizando um conjunto de dados original e confidencial, e está acessível publicamente por meio de uma API. Esse modelo recebe imagens como entrada e gera rótulos como saída. À direita, é mostrado o processo do ataque de extração, no qual é criado um conjunto de dados falso. Nesse processo, INA são enviadas para a API do modelo-alvo e são rotuladas com os RR. Por fim, a rede *Copycat* é treinada com o conjunto de dados falso (CORREIA-SILVA et al., 2018). Imagem de (CORREIA-SILVA et al., 2018).

Embora o uso de imagens aleatórias simplifique a obtenção do conjunto de dados de ataque, ele exige um volume maior de imagens e, consequentemente, mais consultas ao modelo-alvo para atingir uma acurácia similar à do modelo original. Isso ocorre porque não há garantia de que o modelo-alvo produzirá todos os rótulos necessários para o treinamento completo do modelo substituto (CORREIA-SILVA, 2023). No entanto, essa abordagem tem-se mostrado eficaz para replicar o desempenho de modelos, mesmo sem o acesso direto aos dados de treinamento ou aos seus parâmetros.

A união de múltiplos modelos extraídos pode resultar em soluções mais robustas e precisas. Ao combinar diferentes modelos com conhecimentos de diversas fontes, é possível ampliar a capacidade de lidar com múltiplas tarefas e melhorar a acurácia nas inferências. Essa estratégia não só potencializa o desempenho, mas também possibilita um aprendizado contínuo e incremental, alinhado com os objetivos de aprimoramento constante no campo do aprendizado de máquina.

2.7 União de Modelos e Continual Learning

A união de modelos refere-se a técnicas que combinam múltiplos modelos independentes com o objetivo de melhorar o desempenho preditivo ou ampliar as capacidades gerais de um sistema de aprendizado.

Uma estratégia eficaz para unir modelos é o *Ensemble Learning*, que combina as decisões de múltiplos modelos para gerar uma predição final mais robusta. Essa combinação, frequentemente realizada por meio de métodos como votação ou média ponderada, tende a superar a precisão de modelos individuais, pois os erros de cada classificador podem se compensar quando há diversidade suficiente entre eles (DIETTERICH, 2000).

O *Ensemble Learning* também oferece vantagens estatísticas, especialmente em contextos com dados de treinamento limitados em relação ao espaço de hipóteses do modelo. Nesses casos, o algoritmo pode identificar várias hipóteses com desempenho semelhante nos dados de treinamento, mas com variações em sua capacidade de generalização, influenciadas, por exemplo, pela inicialização aleatória dos pesos do modelo. Ao combinar esses modelos em um conjunto e calcular uma predição agregada, o risco de escolher uma hipótese inadequada é reduzido, resultando em uma aproximação mais precisa da função subjacente (DIETTERICH, 2000; FURLANELLO et al., 2018).

Neste contexto de união de modelos, onde o objetivo é aumentar a acurácia e o conhecimento combinados, surge o desafio do *catastrophic forgetting*, ou esquecimento catastrófico (KIRKPATRICK et al., 2017), que ocorre quando novos aprendizados substituem o conhecimento previamente adquirido, levando à perda de acertos em tarefas anteriores. Para enfrentar esse problema, o CL, ou aprendizado contínuo, se destaca como uma abordagem que permite que modelos de IA aprendam e se adaptem continuamente, integrando novos dados e tarefas sem comprometer a precisão em tarefas já dominadas. Essa abordagem é essencial em cenários dinâmicos, onde os dados e os objetivos evoluem constantemente, exigindo modelos capazes de manter o equilíbrio entre aprender o novo e preservar o antigo.

O KD também pode ser utilizado na fusão de modelos, uma abordagem essencial em cenários de CL. Segundo Hou et al. (2018), o processo de aprendizado pode ser comparado ao de um estudante que precisa absorver gradualmente o conhecimento de diferentes disciplinas, sem esquecer o que foi aprendido anteriormente. Nessa analogia, aprender com um *teacher model* é mais eficiente do que estudar diretamente de materiais brutos. Ademais, a revisão periódica dos conhecimentos antigos é fundamental para evitar o esquecimento. Motivada por essa perspectiva, a combinação de Destilação e Retrospecção é proposta para equilibrar a preservação do desempenho em tarefas antigas com a adaptação a novas tarefas.

A integração da destilação do conhecimento com técnicas de *Replay-based* é especial-

mente eficaz em cenários de CL. Conforme discutido por (WANG et al., 2024), a repetição de experiências passadas pode ser naturalmente combinada com o KD, permitindo que o modelo não apenas assimile novas informações, mas também mantenha um desempenho robusto em tarefas previamente aprendidas. Essa abordagem integrada otimiza o processo de aprendizado incremental, garantindo que o conhecimento acumulado seja preservado enquanto novas informações são incorporadas de forma eficiente.

Portanto, algumas técnicas foram desenvolvidas para aplicar o CL nos modelos: A *Elastic Weight Consolidation* (ECW) é um algoritmo que desacelera o aprendizado em certos pesos com base em quão importantes eles são para tarefas vistas anteriormente. A EWC pode ser usada em problemas de aprendizado supervisionado e aprendizado por reforço para treinar várias tarefas sequencialmente sem esquecer as mais antigas (KIRKPATRICK et al., 2017).

Outra abordagem é a *Replay-Based Approach* ou *Replay* de Exemplos, que armazena um subconjunto dos dados antigos e revisita esses exemplos durante o treinamento de novas tarefas. Isso ajuda a mitigar o esquecimento, mantendo o modelo "lembrando" de dados antigos enquanto aprende os novos (WANG et al., 2024)

A técnica de *Progressive Neural Networks* permite a expansão da arquitetura do modelo à medida que novas tarefas e dados são apresentados. Diferente do *fine-tuning*, ela mantém modelos previamente treinados, aprendendo conexões laterais que reutilizam características úteis para novas tarefas. Isso facilita a transferência de conhecimento, evita o catastrophic forgetting e torna a abordagem eficaz para o aprendizado contínuo (RUSU et al., 2022).

O CL proporciona soluções mais integradas e dinâmicas para consolidar modelos. A escolha da abordagem ideal depende do contexto, incluindo os recursos disponíveis, os objetivos de desempenho e a natureza dos dados utilizados.

3 Metodologia

Para consolidar o conhecimento proveniente de múltiplos modelos em um único modelo final da forma mais eficaz, este trabalho propõe um método baseado na extração de conhecimento em cenários *black-box* e na aplicação de técnicas de aprendizado contínuo (CL). A proposta consiste em, inicialmente, extrair os comportamentos de modelos por meio da técnica *Copycat CNN*, utilizando um conjunto amplo de imagens não rotuladas para replicar suas predições sem acesso direto aos dados originais. Em seguida, os conhecimentos obtidos a partir de diferentes domínios são integrados por algumas abordagens, incluindo concatenação da extração de conhecimento, composição por *ensemble* e destilação de conhecimento (KD).

O processo metodológico foi estruturado nas seguintes etapas: (a) seleção de conjuntos de dados públicos relacionados à classificação de imagens; (b) treinamento de modelos oráculos com RNCs adequadas a cada conjunto selecionado; (c) execução da técnica *Copycat CNN* para geração de *fakes datasets* a partir do ImageNet; (d) aplicação das abordagens propostas para unificação de conhecimento, com uso de CL e KD; e (e) avaliação do desempenho dos modelos extraídos em comparação aos oráculos originais. Essas estratégias visam criar modelos *copycats* que sejam capazes de preservar e unificar o conhecimento de modelos *black-box*, de domínios distintos, porém semelhantes, de forma a mitigar o esquecimento catastrófico e maximizar a capacidade de generalização.

3.1 Conjuntos de dados

Foram selecionados dois grupos de conjuntos de dados com características complementares, a fim de explorar a capacidade dos modelos extraídos em aprender e integrar domínios similares, mas com diferentes complexidades visuais e estruturais. A escolha foi motivada pela necessidade de avaliar a técnica proposta tanto em cenários de baixa complexidade quanto em cenários mais desafiadores, variando atributos como número de classes, variação visual e estrutura das imagens, e semelhança entre os domínios.

O primeiro grupo foi composto pelos conjuntos MNIST e SVHN, ambos destinados à classificação de dígitos de 0 a 9, totalizando 10 classes cada. Apesar de pertencerem ao mesmo domínio conceitual (dígitos numéricos), esses conjuntos apresentam diferenças significativas: o MNIST é formado por imagens em tons de cinza de dígitos manuscritos centralizados e normalizados, representando um cenário visual mais limpo e controlado; já o SVHN contém dígitos coloridos extraídos de fotografias de placas de endereços, incluindo variações de iluminação, ruído de fundo e presença de múltiplos dígitos na mesma imagem, tornando o problema mais complexo. Essa diferença estrutural permite avaliar como o

modelo lida com domínios semelhantes em conceito, mas distintos em aparência.

O segundo grupo consistiu no dataset Office-Home, voltado à classificação de 65 objetos, número significativamente superior ao dos conjuntos do primeiro grupo, o que aumenta a complexidade da tarefa. O Office-Home é dividido em quatro domínios: *Art*, *Clipart*, *Product* e *Real*. Neste trabalho, foram utilizados apenas os domínios *Art* e *Real*, por apresentarem maior similaridade visual, mas ainda assim exibirem diferenças relevantes. O domínio *Art* contém ilustrações artísticas estilizadas, enquanto o *Real* apresenta fotografias realistas dos mesmos objetos. Essa configuração permite investigar a capacidade dos modelos de transferir conhecimento entre representações visuais diferentes de uma mesma classe.

Para cada grupo, também foram gerados conjuntos dos dados originais concatenados. No primeiro grupo, as imagens de MNIST e SVHN foram concatenadas em ordem aleatória. No segundo, foi feita a concatenação das imagens de Office-Home-Art e Office-Home-Real. Esses conjuntos combinados foram utilizados para treinar modelos oráculos que serviram como referência em um cenário hipotético no qual os dados de ambos os domínios estariam disponíveis, permitindo comparação direta com os modelos *copycat* obtidos por extração.

Por fim, o conjunto de dados ImageNet foi utilizado como fonte de INA não rotuladas, posteriormente classificadas por cada oráculo para gerar os *fakes datasets*. A escolha do ImageNet se justifica por sua ampla diversidade visual e escala, com mais de 1,2 milhão de imagens distribuídas em suas 1000 classes, o que aumenta a probabilidade de cobrir uma grande variedade de padrões visuais relevantes para diferentes domínios. Essa característica permite simular de forma realista um cenário de ataque *black-box*, no qual um adversário busca replicar o comportamento do modelo-alvo a partir de imagens variadas e não relacionadas diretamente aos dados originais, conforme a abordagem *Copycat CNN*.

3.2 Modelos oráculos

Foram treinados quatro modelos oráculos para os experimentos, simulando um cenário *black-box*, onde modelos treinados são disponibilizados na Internet por meio de APIs com acesso pago, sem que seus dados de treinamento originais sejam fornecidos.

Os dois primeiros, voltados aos conjuntos MNIST e SVHN, foram implementados utilizando a arquitetura ResNet-18, conhecida por seu bom desempenho mesmo em tarefas mais complexas como o ImageNet (HE et al., 2015). Já os modelos treinados com Office-Home-Art e Office-Home-Real utilizaram a arquitetura ResNet-50, devido à maior complexidade e número de classes do problema.

Cada modelo oráculo foi treinado exclusivamente com seu respectivo conjunto de dados, de forma independente, durante 10 épocas, o que se mostrou suficiente para

atingir uma performance considerada satisfatória nos conjuntos de teste. O treinamento foi realizado utilizando a função de perda *CrossEntropyLoss*, com o otimizador *Stochastic Gradient Descent (SGD)* com *momentum* de 0,9 e taxa de aprendizado (*learning rate*) de 1×10^{-3}.

A Tabela 1 apresenta a quantidade de imagens utilizadas para o treinamento e teste de cada modelo oráculo.

Tabela 1 – Detalhes sobre o treinamento dos oráculos: número de imagens utilizadas no treinamento e teste dos modelos oráculos.

Conjunto de Dados	Arquitetura	Imagens Treinamento	Imagens Teste
MNIST	ResNet-18	60.000	10.000
SVHN	ResNet-18	73.257	26.032
Office-Home-Art	ResNet-50	1920 (80% do dominio)	507 (20% do dominio)
Office-Home-Real	ResNet-50	3459 (80% do dominio)	898 (20% do dominio)

3.3 Extração de conhecimento com Copycat CNN

A técnica de extração *Copycat CNN* (CORREIA-SILVA, 2023) foi escolhida por se mostrar uma abordagem eficiente para replicar o conhecimento de modelos em cenários *black-box*, especialmente quando não há acesso às imagens do domínio original.

Diferentemente de métodos que dependem de amostras reais do mesmo domínio do problema, a técnica *Copycat CNN* permite utilizar INA, como as do ImageNet, para induzir o modelo-alvo a fornecer predições, que podem então ser capturadas e utilizadas para treinar um modelo substituto. Essa característica a torna particularmente adequada para o contexto deste trabalho, no qual o objetivo é reconstruir o comportamento dos modelos oráculos, preservando a restrição de ausência total de dados originais.

Com os oráculos treinados, foi realizada a extração de conhecimento por meio da técnica *Copycat CNN*. Essa abordagem consiste em utilizar imagens não rotuladas, neste caso, provenientes do conjunto ImageNet, para obter os RR dos modelos oráculos. A partir disso, foram construídos conjuntos sintéticos rotulados, denominados *fakes datasets*, que foram armazenados para posterior treinamento dos modelos *copycat*.

Como foram treinados quatro modelos oráculos, foram extraídos quatro *fakes datasets*, cada um correspondente a um oráculo. Cada imagem do conjunto INA foi submetida ao respectivo oráculo, totalizando 1.270.441 RR no processo de extração.

Esses quatro *fakes datasets*, um para cada oráculo (MNIST, SVHN, Office-Home-Art e Office-Home-Real), constituíram a base de dados utilizada na etapa de treinamento dos modelos *copycat*, permitindo replicar os comportamentos observados nos modelos originais sem acesso direto aos dados de treinamento.

3.4 Treinamento dos modelos Copycats

A Figura 9 ilustra o fluxo referente ao domínio de números (MNIST e SVHN), o qual também se aplica ao domínio de objetos (*Office-Home*). Nela, observam-se inicialmente os modelos *Oracles*, caracterizados como *black-box* e treinados diretamente com os *datasets* privados. As linhas coloridas representam as extrações realizadas a partir desses modelos, que resultaram na geração dos *fake datasets*. Por fim, a mesma figura evidencia o treinamento dos modelos *copycats* iniciais, conduzido a partir desses conjuntos de dados sintéticos.

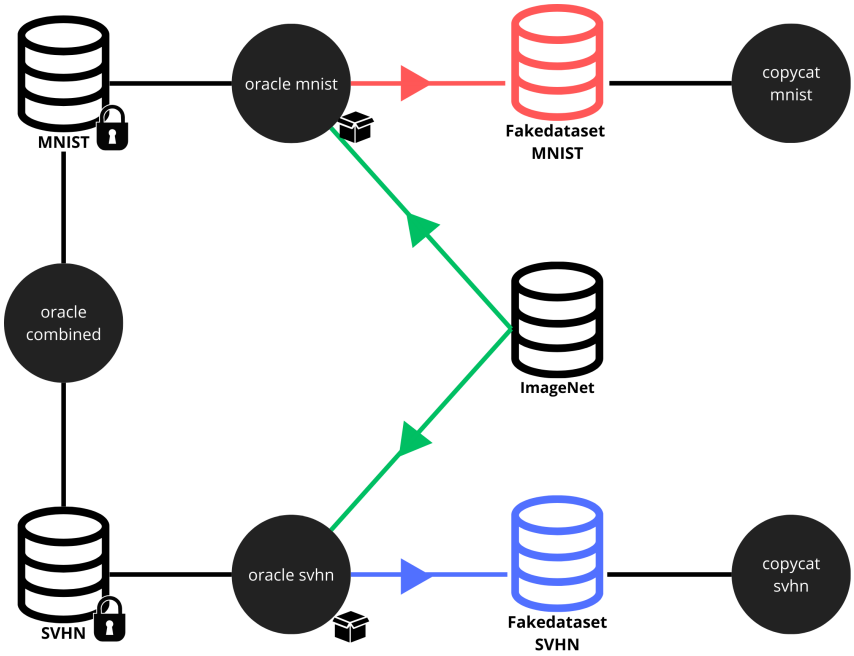

Figura 9 – Geração do cenário de ataque, onde os círculos pretos representam os modelos desenvolvidos durante este trabalho.

Inicialmente, foram treinados quatro modelos *copycat* individuais, cada um correspondente a um oráculo: *copycat-mnist*, *copycat-svhn*, *copycat-art* e *copycat-real*. Para evitar viés arquitetural e isolar os efeitos da extração de conhecimento, cada modelo adotou a mesma arquitetura de seu oráculo correspondente: ResNet-18 para MNIST e SVHN, e ResNet-50 para *Office-Home-Art* e *Office-Home-Real*.

Com o objetivo de aprimorar o desempenho dos modelos nos diferentes domínios, foram investigadas três estratégias complementares. Cada abordagem busca explorar formas distintas de aproveitamento dos *fake datasets* extraídos, resultando em diferentes configurações de modelos *copycat*. A Figura 10 ilustra as três abordagens investigadas: concatenação de dados, ensemble linear e continual learning. Cada uma delas é detalhada a seguir.

Concatenação de dados: nesta abordagem, foram treinados modelos a partir da junção dos *fake datasets* gerados por domínios semanticamente semelhantes. Esse procedimento

resultou nos modelos *copycat-combined* para cada um dos domínios, que compartilham conhecimento entre domínios relacionados por meio da simples fusão de dados. Como ilustrado na Figura 10(a), apresenta o fluxo de treinamento dos modelos *copycat-combined*, obtido a partir da união dos dois *fake datasets* extraídos.

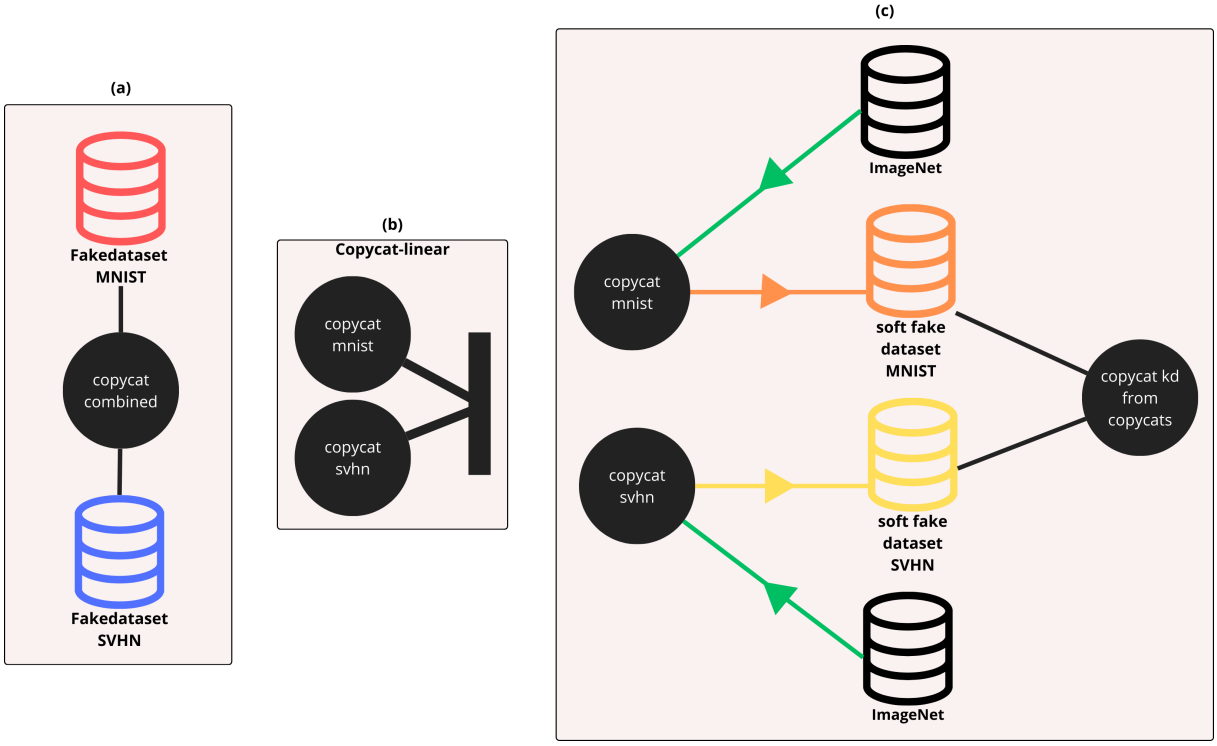

Figura 10 – Fluxo das três abordagens investigadas de modelos *copycat*: (a) concatenação de dados, (b) *ensemble linear* e (c) *knowledge distillation* com as copycats.

Composição por ensemble linear: nesta abordagem, dois modelos *copycats* previamente treinados com os *fake datasets* individuais são executados em paralelo, e suas ativações intermediárias são concatenadas para formar a entrada de uma camada linear adicional. Essa camada final é treinada a partir da fusão dos dois *fake datasets*, integrando o conhecimento extraído de domínios distintos e permitindo a construção dos modelos *copycats-L* para cada domínio. A estratégia segue os princípios de *ensemble learning*, visando gerar predições mais robustas a partir da combinação de múltiplos especialistas (DIETTERICH, 2000). O fluxo dessa composição é ilustrado na Figura 10(b), na qual os modelos *copycats* são unificados pela camada linear.

KD com *copycats*: com o objetivo de unificar o conhecimento de dois domínios distintos em um único modelo, adotou-se uma abordagem de CL baseada em KD (HINTON; VINYALS; DEAN, 2015). Como os oráculos eram tratados como *black-box*, a estratégia foi inspirada nas redes BAN (FURLANELLO et al., 2018), na qual as distribuições de probabilidade (*soft labels*) geradas pelos copycats previamente treinados foram reutilizadas como fonte de conhecimento para um novo modelo. Para isso, foram selecionadas 1 milhão de imagens do ImageNet, rotuladas com os *soft labels* dos copycats de cada domínio (500

mil de cada domínio), e concatenadas em um conjunto sintético utilizado no treinamento do novo modelo *copycat*. Durante o processo de destilação, adotou-se uma temperatura igual a 3, permitindo aprendizado gradual e captura das nuances das distribuições dos modelos originais, em contraste com a temperatura igual a 1, que se assemelha a um treinamento supervisionado tradicional. Essa abordagem preserva o cenário *black-box* dos oráculos e mitiga os efeitos do *catastrophic forgetting*, promovendo uma transferência estável do conhecimento entre os diferentes domínios. Como ilustrado na Figura Figura 10(c), o fluxo do treinamento segue o esquema de aprendizado contínuo com KD.

3.5 Avaliação dos modelos

O desempenho dos modelos *copycat* foi avaliado utilizando os conjuntos de teste originais dos *datasets* MNIST e SVHN. Para os conjuntos Office-Home-Art e Office-Home-Real, selecionou-se aleatoriamente 20% das amostras de cada domínio para compor o conjunto de teste.

A métrica adotada foi a acurácia, que quantifica a proporção de predições corretas, permitindo uma análise objetiva da eficiência dos modelos e de sua capacidade de adaptação entre diferentes domínios.

Os resultados obtidos pelos modelos *copycat* foram comparados com dois grupos de referência:

- **Modelos oráculos originais**: servem como linha de base para mensurar quão próximo o desempenho dos modelos *copycat* se aproxima do modelo-alvo.
- **Modelos treinados com a união dos conjuntos originais** (MNIST+SVHN e Office-Home-Art+Real): representam um cenário idealizado em que o modelo possui acesso aos dados de treinamento dos dois domínios. Esse grupo fornece um limite superior teórico de desempenho (*baseline*) para avaliar a capacidade de integração dos dois domínios.

Essa comparação permitiu verificar a eficácia dos modelos *copycat* na replicação do desempenho dos oráculos em seus próprios conjuntos e na adaptação dos domínios, evidenciando a capacidade de extração e integração de conhecimento dos diferentes métodos testados.

3.6 Configuração de *Hardware* e *Software*

A implementação dos modelos e a execução dos experimentos foram realizadas utilizando a linguagem de programação Python, em conjunto com a biblioteca PyTorch

(PASZKE et al., 2019). Esta biblioteca foi escolhida por sua flexibilidade e ampla aceitação na comunidade científica, além do suporte eficiente à construção e treinamento de redes neurais convolucionais.

Como base para o desenvolvimento dos ataques de extração, foi adotado o framework Copycat CNN[1] , que oferece uma estrutura modular voltada para esse tipo de abordagem. Os scripts de pré-processamento, treinamento, extração e avaliação foram adaptados e estendidos ao longo do projeto, de modo a atender aos objetivos experimentais propostos. O conjunto de dados ImageNet foi utilizado como fonte de imagens para alimentar os ataques de extração, dada sua diversidade e quantidade de amostras.

Todos os experimentos foram conduzidos em um computador disponibilizado pela universidade, com a seguinte configuração de hardware e software:

- **Sistema operacional**: Debian GNU/Linux 12
- **Processador**: Intel Core i7-13700F, com 16 núcleos físicos e 24 threads
- **Memória RAM**: 32 GB
- **Armazenamento**: SSD NVMe de 1 TB
- **Placa de vídeo**: NVIDIA GeForce RTX 4070 com 12 GB de VRAM

[1] <https://github.com/jeiks/copycat_framework>

4 Resultados

Neste capítulo, são apresentados os resultados obtidos a partir da metodologia descrita anteriormente. Inicialmente, discutem-se os resultados da extração dos modelos oráculos para os conjuntos de dados MNIST e SVHN, bem como Office-Home-RealWorld e Office-Home-Art de forma independente. Em seguida, são analisadas as abordagens de união entre domínios e suas limitações, especialmente no que diz respeito à similaridade entre os conjuntos.

4.1 Extração de modelos

Esta seção apresenta a análise quantitativa da acurácia dos modelos copycat extraídos a partir dos oráculos treinados nos conjuntos MNIST e SVHN, utilizando o ImageNet como conjunto de ataque. Os resultados são avaliados considerando tanto a acurácia individual em cada domínio quanto a acurácia média geral. Além disso, é realizada uma comparação entre os modelos copycats e seus respectivos oráculos, incluindo também um modelo adicional utilizado como referência, que representa um cenário ideal, treinado diretamente com os dados originais de ambos os domínios para simular uma união perfeita de conhecimento.

Para cada domínio, foram conduzidos os experimentos conforme descrito na Metodologia. A seguir, são apresentados os resultados de acurácia dos modelos oráculos e dos modelos copycats treinados com os *fakes datasets* gerados a partir dos RR dos oráculos. Inicialmente, são analisados os resultados referentes aos conjuntos MNIST e SVHN; posteriormente, os relativos aos domínios do Office-Home.

4.1.1 Domínios MNIST e SVHN

A Tabela 2 apresenta os resultados obtidos pelos modelos oráculos e pelos respectivos modelos copycat extraídos. Como esperado, os oráculos alcançaram alta acurácia em seus domínios de origem, e para servir de base de comparação, esses modelos também foram testados fora de seu domínio de origem. Enquanto o modelo *oracle-mnist* alcançou 99,1% de acurácia no MNIST, seu desempenho no SVHN foi de 23,3%. Já o *oracle-svhn*, apesar de ter obtido uma acurácia menor em seu domínio (93,3%), obteve desempenho considerável no MNIST (67,3%), resultando em uma média geral superior.

Esse comportamento pode ser atribuído à natureza dos conjuntos de dados. O MNIST é composto por imagens simples, com poucos elementos visuais, o que facilita o aprendizado de padrões específicos, porém menos generalizáveis. Por outro lado, o SVHN

Tabela 2 – Acurácia dos modelos nos datasets de teste do MNIST e SVHN e a média entre eles

Modelos	MNIST	SVHN	Média
oracle-mnist	**99,1%**	23,3%	61,2%
oracle-svhn	67,3%	**93,3%**	80,3%
oracle-combined	99,2%	92,6%	95,9%
copycat-mnist	**98,6%**	23,1%	60,9%
copycat-svhn	66,7%	**92,1%**	79,4%
copycat-combined	95,3%	62,1%	78,7%
copycats-linear	88,2%	75,2%	81,7%
copycat-kd-from-copycats-mnist-svhn	97,3%	69,0%	83,1%

contém imagens mais complexas (com variações de iluminação, angulação, ruído etc.), o que pode ter contribuído para que o modelo treinado nesse domínio desenvolvesse uma representação mais robusta e transferível.

Os modelos copycats extraídos diretamente dos oráculos (*copycat-mnist* e *copycat-svhn*) apresentaram comportamento similar aos seus oráculos correspondentes, com acurácia elevada em seus domínios de origem e menor desempenho no domínio oposto. Como esperado, ambos apresentaram acurácia inferior em relação aos seus respectivos oráculos, o que é característico em cenários de extração sem acesso direto aos dados de treino originais.

O modelo *copycat-combined*, treinado a partir da concatenação dos RR gerados por ambos os oráculos, apresentou um comportamento assimétrico: alcançou alta acurácia no MNIST (95,3%), mas registrou uma queda significativa no SVHN (62,1%). Esse resultado pode estar relacionado à ocorrência de *catastrophic forgetting*, possivelmente influenciada pela diferença entre os domínios. Enquanto o MNIST é composto por imagens em escala de cinza — em que cada pixel possui valores idênticos nos três canais RGB — o SVHN utiliza representações coloridas mais complexas. A integração dessas estruturas distintas pode ter comprometido a retenção simultânea das características específicas de cada domínio.

O modelo *copycats-linear*, que combina os modelos *copycat-mnist* e *copycat-svhn* por meio de uma camada linear adicional, formando uma abordagem de *ensemble*, obteve desempenho mais equilibrado entre os dois domínios: 88,2% no SVHN e 75,2% no MNIST.

Por fim, o modelo *copycat-kd-from-copycats-mnist-svhn*, treinado com os *soft labels* (logits) gerados pelos modelos *copycat-mnist* e *copycat-svhn*, apresentou a maior média de acurácia entre os domínios: 97,3% no MNIST e 69,0% no SVHN, atingindo um desempenho de 85,2% referente ao oracle-combined, como evidenciado na Tabela 3. Isso evidencia o potencial da técnica de KD em cenários de integração de conhecimento de múltiplos domínios.

Tabela 3 – Acurácia relativa dos modelos copycats em relação a maior acurácia dos modelos oráculos no dominio do MNIST e SVHN

Modelos	% ao oracle-mnist	% ao oracle-svhn	% oracle-combined
oracle-mnist	–	25,0%	63,9%
oracle-svhn	67,9%	–	83,7%
oracle-combined	100,1%	99,3%	–
copycat-mnist	99,5%	24,8%	63,5%
copycat-svhn	67,3%	98,7%	82,8%
copycat-combined	96,2%	66,6%	82,1%
copycats-linear	89,0%	80,6%	85,2%
copycat-kd-from-copycats-mnist-svhn	98,1%	69,6%	86,7%

Tabela 4 – Acurácia dos Modelos oráculos e copycats nos datasets de teste de cada dominio e suas médias no dominio Office-Home Art e Real

Modelo	**Real**	**Art**	**Média**
oracle-real	**87,97%**	66,27%	77,12%
oracle-art	76,28%	**78,30%**	77,29%
oracle-combined	86,75%	77,91%	82,33%
copycat-real	**76,28%**	50,49%	63,39%
copycat-art	64,37%	**60,95%**	62,66%
copycat-combined	84,30%	72,58%	78,44%
copycats-art-real-linear	76,84%	61,74%	69,29%
copycat-kd-from-copycats-art-real	76,73%	68,44%	73,70%

4.1.2 Domínios Office-Home Art e RealWorld

Nesta seção, são apresentados os resultados dos testes com os modelos treinados a partir dos domínios *Office-Home RealWorld* e *Office-Home Art*. Como na seção anterior, os modelos oráculos foram inicialmente treinados com os dados reais rotulados de cada domínio, e posteriormente os modelos copycats foram extraídos por meio das técnicas apresentadas na metodologia.

Os resultados de acurácia dos modelos nos conjuntos de teste dos respectivos domínios estão resumidos na Tabela 4. Já a Tabela 5 apresenta a acurácia relativa desses modelos em relação aos seus respectivos oráculos, permitindo uma comparação proporcional de desempenho. Tal como na seção anterior, também foi desenvolvido um modelo adicional de referência, que representa um cenário ideal em que um único modelo é treinado diretamente com os dados originais de ambos os domínios.

Observando a Tabela 4, nota-se que os modelos oráculos apresentam um bom desempenho dentro de seus próprios domínios, com a acurácia de 87,9% para o *oracle-real* e 78,0% para o *oracle-art*. Contudo, assim como observado nos testes com MNIST e SVHN, existe uma assimetria no desempenho dos modelos oráculos em domínios diferentes: o *oracle-real* sofre uma queda acentuada de desempenho ao inferir no domínio Art, enquanto

Tabela 5 – Acurácia relativa dos modelos copycats em relação a acurácia dos modelos oracle-real, oracle-art e oracle-combined

Modelos	% ao oracle-real	% ao oracle-art	% ao oracle-combined
oracle-real	–	84,6%	93,7%
oracle-art	86,7%	–	93,9%
oracle-combined	98,6%	99,5%	–
copycat-real	86,7%	64,5%	77,0%
copycat-art	73,2%	77,8%	76,1%
copycat-combined	95,8%	92,7%	95,3%
copycats-art-real-linear	87,3%	78,9%	84,2%
copycat-kd-from-copycats-art-real	87,2%	87,4%	89,5%

o *oracle-art* é um pouco mais resiliente ao inferir no domínio Real. Essa assimetria pode ser explicada por diferenças visuais entre os estilos de imagem dos domínios (fotos reais vs. arte), sendo o domínio Real mais complexo e detalhado, o que tende a sobreajustar o modelo ao estilo específico de entrada.

No caso dos modelos copycat individuais (*copycat-real* e *copycat-art*), observa-se um comportamento semelhante aos seus respectivos oráculos, porém com perda de desempenho, o que é esperado, já que esses modelos foram treinados apenas com os RR dos oráculos, sem acesso aos dados originais.

O modelo *copycat-combined*, que foi treinado com a concatenação dos dados sintéticos de ambos os oráculos, se destacou ao apresentar a melhor média de acurácia (78,44%) entre os copycats. Seu desempenho foi bastante equilibrado entre os dois domínios, com 84,30% no Real e 72,58% no Art, o que sugere uma boa generalização entre domínios.

O modelo copycats-art-real-linear, que representa a combinação dos copycats individuais por meio da adição de uma camada linear no topo, também apresentou um bom equilíbrio, alcançando acurácias de 76,84% e 61,74% nos domínios RealWorld e Art, respectivamente.

O modelo *copycat-kd-from-copycats-art-real*, que foi treinado com os logits (soft labels) dos modelos copycat individuais,

Apesar de obter uma média ligeiramente inferior à do copycat-combined, seus resultados ainda demonstram um desempenho consistente ao adaptar-se aos dois domínios. Apresentando uma média de 73,7% no dataset de teste, com um desempenho mais equilibrado entre os dois domínios, 76,73% no Real e 68,44% no Art. Isso mostra que o uso de informações mais ricas do que apenas o rótulo hard (como os logits) pode contribuir para uma generalização mais eficaz do conhecimento entre domínios distintos.

Por fim, a Tabela 5 permite avaliar a eficiência dos modelos de forma relativa. Nessa tabela, o modelo copycat-combined alcança 95,3% do desempenho do oracle-combined, o que indica que a extração conjunta dos dois domínios resultou em um desempenho

bastante próximo ao modelo oráculo treinado diretamente com os dados originais de ambos os domínios. De maneira semelhante, o modelo copycat-kd-from-copycats-art-real atinge 89,5% do desempenho do oracle-combined, evidenciando a viabilidade da técnica de destilação com logits na unificação de modelos de domínios semelhantes.

4.2 Discussão

Com base nos resultados apresentados, observa-se que os modelos copycats foram capazes de alcançar desempenhos satisfatórios, aproximando-se dos modelos oráculos originais e o modelo combinado. Esses resultados evidenciam o potencial da técnica de Copycat CNN (CORREIA-SILVA, 2023) na replicação de conhecimento, mesmo sem acesso direto aos dados originais.

Contudo, a eficácia das abordagens variou de acordo com o grau de similaridade entre os domínios envolvidos. No caso do par de domínios Office-Home Art e RealWorld, observou-se que os modelos copycats atingiram bons desempenhos apenas com a concatenação direta das extrações. Isso indica que, como esses domínios compartilham características visuais e estruturais mais próximas, o modelo consegue consolidar o conhecimento sem que haja interferência significativa entre as tarefas. Os conhecimentos extraídos de cada domínio se somam de forma complementar, o que é evidenciado pela alta acurácia do modelo *copycat-combined*, que superou inclusive os oráculos isolados em alguns cenários.

Em contraste, para os domínios MNIST e SVHN, que apresentam diferenças visuais mais acentuadas, a simples concatenação das extrações não foi suficiente. Ao realizar o treinamento contínuo com dados extraídos do MNIST após o SVHN, a acurácia do modelo sofreu uma queda significativa, de 98,7% para 66,6%. Isso indica o fenômeno conhecido como *catastrophic forgetting*, no qual o aprendizado de uma nova tarefa sobrescreve o conhecimento adquirido anteriormente.

Para contornar esse problema, foi necessário empregar técnicas de CL, com destaque para a utilização de KD (HINTON; VINYALS; DEAN, 2015), que permitiu ao modelo copycat assimilar o conhecimento de dois domínios distintos, preservando as características específicas de cada um. Como os oráculos operavam em um cenário *black-box*, ou seja, sem acesso direto às suas arquiteturas ou pesos internos, não era possível obter diretamente suas distribuições de probabilidade. Para superar essa limitação, adotou-se uma abordagem inspirada nas BAN (FURLANELLO et al., 2018), na qual os próprios modelos copycats, previamente treinados com uma extração específica, foram reutilizados como professores no processo de destilação. Essa estratégia possibilitou a utilização de *soft labels* mesmo sem acesso direto aos oráculos, permitindo uma transferência de conhecimento mais rica, contínua e adaptada, sem comprometer o cenário *black-box* original.

Além disso, também foram avaliadas estratégias baseadas em *ensemble learning*

(DIETTERICH, 2000), onde os modelos copycats treinados separadamente para cada domínio eram combinados por meio de uma camada linear final. Essa abordagem foi eficaz em preservar os pontos fortes de cada modelo, oferecendo um equilíbrio entre desempenho e robustez.

Em resumo, os resultados indicam que a escolha da técnica de fusão de conhecimento deve considerar o grau de dissimilaridade entre os domínios. Para domínios mais similares, a concatenação direta é suficiente. Já para domínios mais distintos, abordagens baseadas em CL, distillation ou ensemble tornam-se essenciais para evitar perda de informação e garantir uma integração eficaz dos conhecimentos extraídos.

5 Conclusão

O objetivo geral deste trabalho foi investigar a extração e integração do conhecimento de múltiplos domínios por meio de técnicas de extração de modelos (*model extraction*), considerando as restrições do cenário *black-box* e buscando evitar o *catastrophic forgetting*. Para isso, foram propostas diferentes abordagens para integrar o conhecimento de modelos oráculos especializados, com o intuito de treinar um único modelo capaz de generalizar entre domínios semelhantes, porém com características distintas, sem acesso aos dados originais de treinamento.

Com base nos resultados obtidos, confirma-se que o objetivo foi atingido. Os métodos propostos mostraram-se eficazes na consolidação do conhecimento proveniente de múltiplos modelos *black-box* em um único modelo final, preservando o desempenho nos diferentes domínios e mitigando o *catastrophic forgetting*, ainda que com variações na eficácia entre as abordagens testadas.

A extração de modelos foi conduzida com êxito em diferentes domínios — MNIST, SVHN, Office-Home Art e RealWorld — utilizando imagens do ImageNet como conjunto de ataque. Os modelos copycats obtidos apresentaram desempenho competitivo em relação aos respectivos oráculos, evidenciando que é possível replicar o comportamento dos modelos originais e aproximar-se do desempenho do modelo "ideal" (oracle-combined), treinado diretamente com os dados originais de ambos os domínios.

A principal contribuição deste trabalho foi a comparação entre diferentes estratégias para a fusão de modelos extraídos: concatenação de dados, ensemble learning e KD com soft labels. Os resultados indicaram que a concatenação funciona bem quando os domínios são semanticamente próximos, como no caso do Office-Home. Já para domínios mais distintos, como MNIST e SVHN, estratégias mais robustas, como o CL com destilação, mostraram-se mais eficazes para mitigar o *catastrophic forgetting* e preservar o conhecimento aprendido.

Como limitações do estudo, destaca-se o alto custo computacional das abordagens mais complexas, como a destilação com soft labels. Outra limitação está na avaliação restrita a domínios visuais; embora o método seja potencialmente generalizável, sua aplicabilidade a outras tarefas permanece como trabalho futuro.

Apesar dos avanços alcançados, considera-se algumas estratégias que poderiam aprimorar a abordagem desenvolvida neste trabalho e que permanecem como oportunidades para estudos futuros, como:

- Usar de técnicas mais sofisticadas de integração, como estratégias baseadas em *Maximum Softmax Probability* (MSP), que auxiliam na identificação de regiões de

baixa confiança do modelo, permitindo um controle mais refinado da fusão entre tarefas distintas e contribuindo para reduzir os efeitos do *catastrophic forgetting* durante o aprendizado contínuo.

- Usar de abordagens baseadas em ataques adversariais consensuais para gerar conjuntos sintéticos com características compartilhadas entre domínios distintos. Nesse cenário, o consenso entre diferentes modelos pode ser utilizado como critério para a geração de dados sintéticos capazes de generalizar simultaneamente sobre os dois domínios considerados.

- Utilizar de forma mais aprofundada as redes BAN, ou a exploração de outras arquiteturas dedicadas ao CL, representa uma direção relevante para melhorar a transferência de conhecimento entre modelos copycats e ampliar a capacidade de adaptação a novos domínios sem comprometer o desempenho nos domínios previamente assimilados.

- Ampliar os testes para uma maior variedade de domínios e oráculos, incluindo domínios visualmente mais divergentes ou com maior número de classes. Isso permitirá avaliar a escalabilidade, robustez e limites das estratégias de extração e integração propostas neste trabalho.

Conclui-se, portanto, que a abordagem adotada neste trabalho contribui para o avanço na área de segurança em aprendizado de máquina, especialmente no contexto de ataques de extração e adaptação contínua de conhecimento. Através da experimentação sistemática e da comparação entre estratégias, este estudo oferece um panorama sólido e replicável sobre como extrair, combinar e preservar conhecimento em ambientes restritivos e com múltiplos domínios. Além disso, ao evidenciar a viabilidade técnica da integração de modelos extraídos, os resultados reforçam a necessidade de compreender os riscos associados a esse tipo de ataque, em que agentes adversários podem combinar informações provenientes de múltiplas fontes. Essa compreensão é fundamental para o desenvolvimento de estratégias que não apenas dificultem a extração, mas também inibam a fusão de conhecimento entre diferentes domínios, contribuindo para a formulação de mecanismos de defesa mais robustos e garantindo maior proteção à confidencialidade e à integridade dos modelos de IA.

Referências

COPPIN, B. *Artificial Intelligence Illuminated*. Jones and Bartlett Publishers, 2004. (Jones and Bartlett illuminated series). ISBN 9780763732301. Disponível em: <https://books.google.com.br/books?id=LcOLqodW28EC>. Citado na página 16.

CORREIA-SILVA, J. R. *Copycat CNN*: Convolutional neural network extraction attack with unlabeled natural images. Tese (Doutorado) — Curso de Computação, Universidade Federal do Espírito Santo, Vitória, ES, 2023. Disponível em: <https://sappg.ufes.br/tese_drupal//tese_17166_Tese-Jacson-2018142921.pdf>. Citado 6 vezes nas páginas 11, 13, 27, 28, 33 e 42.

CORREIA-SILVA, J. R. et al. Copycat cnn: Stealing knowledge by persuading confession with random non-labeled data. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018. Disponível em: <http://dx.doi.org/10.1109/IJCNN.2018.8489592>. Citado 5 vezes nas páginas 11, 12, 13, 27 e 28.

DIETTERICH, T. G. Ensemble methods in machine learning. In: *Multiple Classifier Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000. p. 1–15. ISBN 978-3-540-45014-6. Citado 4 vezes nas páginas 12, 29, 35 e 43.

DIKE, H. U. et al. Unsupervised learning based on artificial neural network: A review. In: IEEE. *2018 IEEE International Conference on Cyborg and Bionic Systems (CBS)*. [S.l.], 2018. p. 322–327. Citado na página 15.

FRIKHA, A. et al. *Towards Data-Free Domain Generalization*. 2022. Disponível em: <https://arxiv.org/abs/2110.04545>. Citado na página 12.

FURLANELLO, T. et al. *Born Again Neural Networks*. 2018. Disponível em: <https://arxiv.org/abs/1805.04770>. Citado 4 vezes nas páginas 26, 29, 35 e 42.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>. Citado 8 vezes nas páginas 15, 16, 17, 18, 19, 21, 24 e 25.

GOODFELLOW, I. J. et al. *Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks*. 2014. Disponível em: <https://arxiv.org/abs/1312.6082>. Citado na página 19.

HE, K. et al. *Deep Residual Learning for Image Recognition*. 2015. Disponível em: <https://arxiv.org/abs/1512.03385>. Citado 3 vezes nas páginas 10, 20 e 32.

HINTON, G.; VINYALS, O.; DEAN, J. *Distilling the Knowledge in a Neural Network*. 2015. Disponível em: <https://arxiv.org/abs/1503.02531>. Citado 3 vezes nas páginas 26, 35 e 42.

HOU, S. et al. Lifelong learning via progressive distillation and retrospection. In: FERRARI, V. et al. (Ed.). *Computer Vision – ECCV 2018*. Cham: Springer International Publishing, 2018. p. 452–467. ISBN 978-3-030-01219-9. Citado na página 29.

JUNIOR, J. S.; CAURIN, G. A. d. P. *Redes neurais convolucionais aplicadas à detecção de não-conformidades em equipamentos industriais*. 2022. Citado na página 9.

KIRKPATRICK, J. et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, Proceedings of the National Academy of Sciences, v. 114, n. 13, p. 3521–3526, mar. 2017. ISSN 1091-6490. Disponível em: <http://dx.doi.org/10.1073/pnas.1611835114>. Citado 3 vezes nas páginas 12, 29 e 30.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: PEREIRA, F. et al. (Ed.). *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2012. v. 25. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>. Citado na página 10.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 60, n. 6, p. 84–90, may 2017. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/3065386>. Citado 2 vezes nas páginas 19 e 21.

LECUN, Y. et al. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, v. 1, n. 4, p. 541–551, 1989. Citado na página 18.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proc. IEEE*, v. 86, p. 2278–2324, 1998. Disponível em: <https://api.semanticscholar.org/CorpusID:14542261>. Citado 4 vezes nas páginas 9, 10, 18 e 21.

LORENTE Òscar; RIERA, I.; RANA, A. *Image Classification with Classic and Deep Learning Techniques*. 2021. Disponível em: <https://arxiv.org/abs/2105.04895>. Citado na página 9.

MASLEJ, N. et al. *Artificial Intelligence Index Report 2024*. 2024. Disponível em: <https://arxiv.org/abs/2405.19522>. Citado na página 12.

MESQUITA, L. A. F.; UEYAMA, J. *Redes neurais artificiais aplicadas a séries temporais para predição de enchentes*. Dissertação (Mestrado) — Universidade de São Paulo, 2024. Citado na página 9.

MITCHELL, M. H. T. *Machine Learning*. [S.l.]: McGraw-Hill Science/Engineering/Math, 1997. Citado 3 vezes nas páginas 15, 16 e 17.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. *Sistemas inteligentes-Fundamentos e aplicações*, v. 1, n. 1, p. 32, 2003. Citado 2 vezes nas páginas 15 e 21.

NETZER, Y. et al. Reading digits in natural images with unsupervised feature learning. *NIPS*, 01 2011. Citado 2 vezes nas páginas 10 e 22.

OLIVEIRA, V. H. R. et al. *Predição da Idade Cerebral a partir de Imagens de Ressonância Magnética utilizando Redes Neurais Convolucionais*. 2021. Disponível em: <https://arxiv.org/abs/2112.12609>. Citado na página 9.

OPENAI. *ChatGPT: Large Language Model*. 2023. Acesso em: 30 ago. 2025. Disponível em: <https://openai.com/>. Citado na página 11.

OREKONDY, T.; SCHIELE, B.; FRITZ, M. *Knockoff Nets: Stealing Functionality of Black-Box Models*. 2018. Disponível em: <https://arxiv.org/abs/1812.02766>. Citado 4 vezes nas páginas 11, 12, 13 e 27.

PAN, S. J.; YANG, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, IEEE, v. 22, n. 10, p. 1345–1359, 2009. Citado 2 vezes nas páginas 10 e 24.

PASZKE, A. et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. Disponível em: <https://arxiv.org/abs/1912.01703>. Citado na página 37.

RUSSAKOVSKY, O. et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, v. 115, n. 3, p. 211–252, 2015. Citado 2 vezes nas páginas 10 e 21.

RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach, Global Edition*. Pearson Education, 2021. ISBN 9781292401171. Disponível em: <https://books.google.com.br/books?id=cb0qEAAAQBAJ>. Citado na página 15.

RUSU, A. A. et al. *Progressive Neural Networks*. 2022. Disponível em: <https://arxiv.org/abs/1606.04671>. Citado 2 vezes nas páginas 12 e 30.

SAMUEL, A. L. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.*, IBM Corp., USA, v. 3, n. 3, p. 210–229, jul. 1959. ISSN 0018-8646. Disponível em: <https://doi.org/10.1147/rd.33.0210>. Citado na página 15.

SIMONYAN, K.; ZISSERMAN, A. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. Disponível em: <https://arxiv.org/abs/1409.1556>. Citado 2 vezes nas páginas 10 e 20.

SRINIVAS, S. et al. A taxonomy of deep convolutional neural nets for computer vision. *Frontiers in Robotics and AI*, v. 2, 2016. ISSN 2296-9144. Disponível em: <https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2015.00036>. Citado na página 10.

SZEGEDY, C. et al. *Going Deeper with Convolutions*. 2014. Disponível em: <https://arxiv.org/abs/1409.4842>. Citado na página 20.

SZEGEDY, C. et al. *Rethinking the Inception Architecture for Computer Vision*. 2015. Disponível em: <https://arxiv.org/abs/1512.00567>. Citado na página 18.

TAN, C. et al. *A Survey on Deep Transfer Learning*. 2018. Disponível em: <https://arxiv.org/abs/1808.01974>. Citado na página 26.

TOUVRON, H. et al. *LLaMA: Open and Efficient Foundation Language Models*. 2023. Disponível em: <https://arxiv.org/abs/2302.13971>. Citado na página 11.

TRAMèR, F. et al. *Stealing Machine Learning Models via Prediction APIs*. 2016. Disponível em: <https://arxiv.org/abs/1609.02943>. Citado 4 vezes nas páginas 10, 11, 13 e 27.

VENKATESWARA, H. et al. Deep hashing network for unsupervised domain adaptation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2017. p. 5018–5027. Citado 3 vezes nas páginas 22, 23 e 25.

WANG, L. et al. *A Comprehensive Survey of Continual Learning: Theory, Method and Application*. 2024. Disponível em: <https://arxiv.org/abs/2302.00487>. Citado 2 vezes nas páginas 12 e 30.

WEISS, K.; KHOSHGOFTAAR, T. M.; WANG, D. A survey of transfer learning. *Journal of Big Data*, v. 3, n. 1, p. 9, May 2016. ISSN 2196-1115. Disponível em: <https://doi.org/10.1186/s40537-016-0043-6>. Citado 2 vezes nas páginas 10 e 11.

YOSINSKI, J. et al. *How transferable are features in deep neural networks?* 2014. Disponível em: <https://arxiv.org/abs/1411.1792>. Citado 3 vezes nas páginas 24, 25 e 26.

ZHANG, A. et al. *Dive into Deep Learning*. [S.l.]: Cambridge University Press, 2023. <https://D2L.ai>. Citado 3 vezes nas páginas 18, 19 e 20.

ZHOU, Z.; LIU, S. *Machine Learning*. Springer Nature Singapore, 2021. ISBN 9789811519666. Disponível em: <https://books.google.com.br/books?id=Zd5hywEACAAJ>. Citado 2 vezes nas páginas 16 e 17.