Pós-Graduação em Ciência da Computação

Miguel Domingos de Santana Wanderley

Transferindo Conhecimento de Textos para Imagens Através da Aprendizagem das Características Semânticas

Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
http://cin.ufpe.br/~posgraduacao

Recife
2018

Miguel Domingos de Santana Wanderley

Transferindo Conhecimento de Textos para Imagens Através da Aprendizagem das Características Semânticas

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre Acadêmico em Ciência da Computação.

Área de Concentração: Inteligência Computacional

Orientador: Prof. Ricardo Bastos Cavalcante Prudêncio

Recife
2018

Catalogação na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

W245t Wanderley, Miguel Domingos de Santana
Transferindo conhecimento de textos para imagens através da aprendizagem das características semânticas / Miguel Domingos de Santana Wanderley. – 2018.
81 f.: il., fig.

Orientador: Ricardo Bastos Cavalcante Prudêncio.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2018.
Inclui referências e apêndices.

1. Inteligência artificial. 2. Redes neurais. I. Prudêncio, Ricardo Bastos Cavalcante (orientador). II. Título.

006.3 CDD (23. ed.) UFPE- MEI 2018-133

Miguel Domingos de Santana Wanderley

Transferindo Conhecimento de Textos para Imagens Através da Aprendizagem das Características Semânticas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação

Aprovado em: 21/08/2018.

BANCA EXAMINADORA

_____

Prof. Dr. Tsang Ing Ren
Centro de Informática/UFPE

_____

Prof. Dr. Filipe Rolim Cordeiro
Departamento de Estatística e Informática/UFRPE

_____

Prof. Dr. Ricardo Bastos Cavalcante Prudêncio
Centro de Informática / UFPE
(Orientador)

Dedico este trabalho a Maria Luíza Amaral e Josefa Elza de Santana,
por todo amor e carinho, aos quais serei eternamente grato.

AGRADECIMENTOS

Agradeço à minha família e à minha namorada, todo o amor, preocupação, paciência e motivação nos momentos difíceis. Agradeço também ao meu orientador todo apoio, suporte e orientação dada ao longo desses anos. Agradeço meus professores e colegas de trabalho com quem aprendi, trabalhei e dividi conquistas. Sem todos os mencionados, este trabalho não seria possível.

RESUMO

Redes neurais profundas vem mostrando um expressivo desempenho em tarefas de reconhecimento de imagens. Dentre as principais técnicas de redes neurais profundas, destacam-se as redes neurais convolucionais, as quais apresentam a capacidade de aprender características de alto nível em imagens, considerando o aspecto espacial das mesmas. A profundidade das redes neurais convolucionais permite que características de baixo nível sejam combinadas em características de mais alta complexidade, gradativamente, até que imagens possam ser codificadas em características de alto nível. Dentre as atividades de reconhecimento de imagens podemos mencionar a classificação de imagens, detecção de objetos e segmentação de imagens. No entanto, as principais técnicas de redes convolucionais profundas demandam volumes massivos de imagens rotuladas para treinamento, nem sempre disponíveis. Neste contexto, técnicas de transferência de conhecimento vem sendo adotadas para superar a falta de dados rotulados disponíveis para treinamento de modelos em tarefas específicas. De modo geral, transferência de aprendizagem busca utilizar dados disponíveis em quantidades expressivas em um determinado domínio fonte para possibilitar uma aprendizagem mais eficiente de um modelo em dados de um domínio alvo, geralmente mais escasso. Este trabalho apresenta uma nova arquitetura de rede neural profunda com a capacidade de transferir conhecimento de dados textuais associados a imagens (domínio fonte) para auxiliar na atividade de reconhecimento de imagens (domínio alvo). Como componentes a rede proposta utiliza um extrator convolucional de características visuais latentes de imagens (codificador) enquanto um modelo generativo probabilístico é usado para definir tópicos semânticos textuais. Uma combinação de classificadores é então utilizada para estimar tópicos semânticos para novas instâncias de imagens baseada nas características visuais latentes desta instância. Experimentos foram conduzidos para avaliar o quão relacionadas estão as características latentes em ambos os domínios (textual e visual) e ainda verificar a eficácia dos tópicos semânticos preditos pelo modelo proposto na tarefa de classificação de imagens. Resultados promissores foram verificados comparando-se com diferentes abordagens estado da arte neste cenário multimodal heterogêneo.

Palavras-chaves: Redes Neurais Profundas. Transferência de Conhecimento. Domínios Heterogêneos.

ABSTRACT

Deep neural networks have been showing significant performance in image recognition tasks. Among the main techniques of deep neural networks, we highlight the convolutional neural networks, which present the ability to learn high-level features from images, considering the spatial aspect of them. The depth of convolutional neural networks allows low-level features to be combined into features of higher complexity, gradually, until images can be encoded into high-level features. Among the image recognition tasks, we can mention the image classification, objects detection, and images segmentation. However, the main techniques of deep convolutional networks require massive volumes of labeled images for training, not always available. In this context, knowledge transfer techniques have been adopted to overcome the lack of labeled data available for training models for specific tasks. In general, transfer learning seeks to use available data in significant quantities in a particular source domain to enable a more efficient learning of a model in data from a target domain, generally more scarce. This work presents a new deep neural network architecture with the ability to transfer knowledge of textual data (source domain) associated with images (target domain) to assist in image recognition tasks. The proposed network uses as components a convolutional feature extractor (encoder) of latent visual image characteristics, while a generative probabilistic model is used to learn textual semantic topics. An ensemble of classifiers is then used to estimate semantic topics for new instances of images, based on the latent visual features of the test instance. Experiments were conducted to evaluate how related are the embedded features in both domains (textual and visual) and to verify the efficacy of the semantic topics predicted by the proposed model in image classification tasks. Promising results were verified comparing with different state-of-the-art approaches in this heterogeneous multimodal scenario.

Key-words: Deep Learning. Transfer Learning. Heterogeneous Domains.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE SIGLAS

ADALINE	*Adaptive Linear Neuron* (Neurônio Adaptativo Linear)
CAE	*Convolutional Autoencoder* (Auto Codificadores Convolucionais)
CNN	*Convolutional Neural Network* (Rede Neural Convolucional)
DBN	*Deep Belief Network*
DTN	*Deep Transfer Networks* (Redes de Transferência Profunda)
EM	*Expectation–Maximization*
GDE	*Gradiente Descendente Estocástico*
HTL	*Heterogeneous Transfer Learning for Image Classification* (Transferência de Aprendizagem Heterogênea para Classificação de Imagens)
ILSVRC	*ImageNet Large Scale Visual Recognition Challenge*
MLP	*Multilayer Perceptron*
PM	Perceptron Multicamadas
RBF	*Radial Basis Function* (Função de Base Radial)
RBM	*Restricted Boltzmann Machine*
ReLU	*Rectified Linear Unit*
RGB	*Red* (vermelho), *Green* (verde) e *Blue* (azul)
SAE	*Stacked Auto-Encoders* (Auto Codificadores Empilhados)
SVM	*Support Vector Machine* (Máquina de Vetores de Suporte)
TTI	*Translator from Texts to Images* (Tradutor de Textos para Imagens)

LISTA DE SÍMBOLOS

Λ	Modelo generativo tradutor de espaço textual para espaço semântico
λ	Hiper-parâmetro que define o tamanho do espaço semântico considerado pelo modelo proposto
$\mathbb{T}$	Espaço textual original – vocabulário
T	Tamanho do vocabulário
$\mathbf{t}$	Vetor de características textuais
t^i	i-ésimo termo textual
$\mathbb{T}_s$	Espaço textual reduzido
$\mathbf{t}_s$	Vetor de características textuais reduzido
$\mathbb{S}$	Espaço semântico
$\mathbf{s}$	Vetor de características semânticas
s^i	i-ésimo termo semântico
$\mathcal{X}$	Espaço de características visuais
$\mathbf{x}$	Vetor de características visuais
$\mathcal{D}$	Conjunto de pares de imagens e textos associados
$\mathbb{C}$	Conjunto de classificadores binários de tópicos semânticos
$\mathbb{G}$	Conjunto de modelos generativos textuais

SUMÁRIO

1 INTRODUÇÃO

Neste capítulo introdutório, apresentamos o contexto deste trabalho, que é no campo da Transferência de Conhecimento em Aprendizagem de Máquina. Depois disso, estabelecemos o problema de interesse no campo mais específico da Transferência Heterogênea de Aprendizagem e Aprendizagem Profunda. Em seguida, definimos os objetivos deste estudo e as contribuições alcançadas. Finalmente, apresentamos um esboço dos assuntos tratados nos próximos capítulos.

1.1 Contexto

Técnicas de Aprendizagem de Máquina necessitam de dados para treinar modelos que são aplicados para desempenhar determinadas tarefas em diferentes campos de aplicações. No caso das técnicas supervisionadas de Aprendizagem de Máquina, ainda se faz necessária a existência de dados marcados, ou rotulados, para possibilitar a aprendizagem dos modelos de interesse.

Um dos domínios de aplicação da Aprendizagem de Máquina é a Visão Computacional, que combina técnicas de aprendizagem e de processamento digital de imagens. Igualmente, para as técnicas de Aprendizagem de Máquina voltadas para a Visão Computacional, existe uma necessidade de dados para treino. Dentre as principais aplicações de Visão Computacional podemos mencionar a classificação de imagens, segmentação e detecção de objetos; todas essas são atividades tratadas por algoritmos de treinamento supervisionados, que do mesmo modo demandam dados marcados para treino.

Técnicas recentes de Aprendizagem Profunda vêm apresentando ótimos desempenhos em aplicações de Visão Computacional, sejam estas, tarefas de classificação de imagens, segmentação, detecção de objetos, extração de características visuais, entre outras (KRIZHEVSKY; SUTSKEVER; HINTON, 2012; SIMONYAN; ZISSERMAN, 2014; HE et al., 2016; HOWARD et al., 2017; IANDOLA et al., 2016; GOODFELLOW et al., 2016).

1.1.1 Contexto Geral

Conforme mencionado, técnicas de Aprendizagem de Máquina demandam dados para treinamento de modelos, fazendo com que o desempenho dos modelos treinados estejam fortemente relacionados com a qualidade dos dados utilizados para treinamento. Espera-se que os dados utilizados para treino sejam uma amostra representativa dos dados existentes no domínio real para o qual pretende-se aplicar os modelos treinados.

No contexto das técnicas de Aprendizagem Profunda, a necessidade por dados para treino é ainda mais crítica. As principais arquiteturas de Redes Neurais Profundas usual-

mente demandam um volume expressivo de dados para treino para alcançar bons resultados, na ordem de centenas até centenas de milhares de instâncias.

Apesar do sucesso e do bom desempenho das Redes Neurais Profundas, aplicações reais usualmente apresentam fortes limitações de disponibilidade de dados suficientes para treino, dificultando o treinamento de modelos igualmente robustos em aplicações reais.

A limitação de dados suficientemente disponíveis para treinamento de modelos não é uma particularidade apenas das técnicas em Aprendizagem Profunda. Este é um problema previamente existente em Aprendizagem de Máquina, o qual busca ser resolvido por técnicas de Transferência de Aprendizagem (PAN; YANG et al., 2010).

Em linhas gerais, o problema em questão pode ser definido como a indisponibilidade de dados suficientes para treinamento de modelos em um determinado contexto de aplicação. Para a solução deste problema, a Transferência de Aprendizagem sugere utilizar dados disponíveis em um contexto menos escasso, com características similares ao contexto da aplicação de interesse. Ao utilizar procedimentos de Transferência de Aprendizagem, busca-se beneficiar um modelo em um contexto com dados escassos com informações provenientes de outro contexto, com dados em maior disponibilidade, para realizar uma tarefa nova porém similar (PAN; YANG et al., 2010).

Existem diferentes abordagens de Transferência de Aprendizagem. As mais antigas buscam resolver o problema da variação da distribuição entre os dados de diferentes contextos utilizando abordagens estatísticas. A temática passou a ser cada vez mais explorada e investigada em diversos estudos, demonstrando a crescente relevância da área e a recorrência do problema de falta de dados para treino em contextos específicos (LU et al., 2015).

No campo da Aprendizagem Profunda, a Transferência de Conhecimento se tornou ainda mais explorada por conta da capacidade de se reutilizar modelos pré-treinados em bases de dados mais abundantes em problemas mais específicos, contornando, assim, a falta de dados em volumes expressivos para treino dos modelos (YOSINSKI et al., 2014; OQUAB et al., 2014; SOEKHOE; PUTTEN; PLAAT, 2016; ZEILER; FERGUS, 2014).

1.1.2 Contexto Específico

A área da Transferência de Conhecimento vem apresentando uma crescente evolução, apresentando diversas soluções para diferentes manifestações do problema da indisponibilidade de dados entre contextos relacionados. Os problemas tratados pela Transferência de Conhecimento variam de acordo com as variações de tarefas a serem realizadas, diferença entre as distribuições dos dados, diferença entre os espaços de características, entre outros. Transferência de Conhecimento apresenta-se como uma área cada vez mais ampla e relevante, despertando interesses de pesquisas em diversas frentes de atuação (WEISS; KHOSHGOFTAAR; WANG, 2016).

Um dos problemas abordados é a necessidade de aprender informações entre domínios de origens diferentes, ou seja, domínios com espaços de características distintos. Este contexto específico busca ser tratado por técnicas de Transferência Heterogênea de Aprendizagem (WEISS; KHOSHGOFTAAR; WANG, 2016). No entanto, esta aplicação particular atualmente apresenta-se como uma área igualmente relevante, mas pouco explorada, dada a dificuldade em relacionar domínios distintos (WEISS; KHOSHGOFTAAR; WANG, 2016).

Uma das aplicações no problema de relacionar domínios distintos é a aprendizagem de informações presentes entre dados textuais e visuais. Esta categoria de aplicações ganhou força com a crescente disponibilidade de dados visuais (imagens) e dados textuais (*Tags*, descrições, legendas e afins) associados presentes em portais na *web*, páginas de compartilhamento de imagens e plataformas digitais de redes sociais.

Alguns trabalhos apresentam técnicas para aprendizagem entre domínios textuais e visuais de dados disponíveis em páginas *web*, denotando a relevância desta linha de pesquisa e a motivação em aprender as informações latentes que relacionam os dois domínios (ZHU et al., 2011; QI; AGGARWAL; HUANG, 2011; ZHANG et al., 2015; SHU et al., 2015).

Apesar do sucesso das Redes Neurais Profundas em atividades de Visão Computacional, encontramos poucos trabalhos que utilizam desta abordagem pra tratar do problema de aprendizagem entre os domínios textuais e visuais (ZHANG et al., 2015; SHU et al., 2015). As arquiteturas propostas por Zhang et al. (2015) e Shu et al. (2015) realizam a aprendizagem do espaço latente de forma implícita ao modelo e representam um dos poucos trabalhos que utilizam redes profundas para aprender mutuamente entre os domínios textuais e visuais.

Umas das direções em aberto na área é a proposta de um modelo de Rede Neural Profunda que aprenda de forma mais explícita informações latentes (provavelmente semânticas) entre os domínios textuais e visuais, conforme sugerido por Qi, Aggarwal e Huang (2011) em sua abordagem não profunda. Esta é a direção mais específica explorada neste trabalho, no contexto específico de aprender informações semânticas entre dados textuais e visuais.

1.2 Objetivos

Este trabalho tem a motivação de propor uma técnica para realizar a transferência de conhecimento entre domínios heterogêneos (domínio textual e domínio visual), mais precisamente, uma abordagem capaz de aprender explicitamente as informações semânticas que correlacionam textos e imagens de um mesmo fato do mundo real. Complementando a motivação do trabalho, buscamos definir uma técnica beneficiada pela Aprendizagem Profunda, diante dos recentes bons resultados apresentados em tarefas relacionadas com Visão Computacional e extração de características visuais de alto nível.

Diante da motivação apresentada, foram traçados um objetivos geral e objetivos específicos de pesquisa para este trabalho, detalhados nas seções seguintes.

1.2.1 Objetivo Geral

De acordo com o contexto descrito anteriormente, definimos o problema abordado por esse trabalho como sendo a necessidade de possibilitar a aprendizagem de um modelo considerando domínios diferentes, neste caso, os domínios textuais e visuais.

O **objetivo geral** deste trabalho é propor uma arquitetura de Rede Neural Profunda que possa ser treinada com dados textuais e visuais relacionados. Os dados textuais referenciam o domínio alvo do problema, para o qual esperamos que o modelo faça predições. Já os dados visuais referenciam o domínio fonte do problema, entendido como o domínio auxiliar com maior quantidade de dados disponíveis. A finalidade da arquitetura proposta é estimar possíveis dados textuais para novas imagens de teste.

O domínio fonte é entendido como o domínio com dados em maior disponibilidade, no contexto deste trabalho, são os dados visuais, *i.e.*, as imagens sem anotações ou *tags* associadas. Por outro lado, entendemos como domínio alvo, com a característica de ser um domínio com uma quantidade mais limitada de dados, o domínio textual associado às imagens, sendo assim, são os exemplos de pares de imagem e texto associado. É para o domínio de textos associados à descrição de uma imagem que objetiva-se a aplicação da arquitetura proposta. Note que o domínio alvo consiste em textos associados com descrição de imagens, e não textos avulsos desvinculados de imagens.

1.2.2 Objetivos Específicos

Os objetivos específicos delineiam os marcos intermediários do trabalho necessários para a completude do objetivo geral mencionado. Estes objetivos tornam mais claras as atividades realizadas ao longo do trabalho.

- **OE1**: Implementar um protótipo para a técnica proposta e treinar um modelo em uma base de dados relevante para o contexto específico abordado.
- **OE2**: Definir uma regra para definição de traços semânticos em dados textuais.
- **OE3**: Realizar experimentação para avaliar a capacidade de aprendizagem e a qualidade da técnica proposta.
- **OE4**: Comparar a técnica proposta com técnicas similares no mesmo contexto.

1.3 Contribuições

Como principal contribuição este trabalho apresenta uma arquitetura de Rede Neural Profunda para recomendar dados textuais para novas imagens de teste. Para o procedimento de treinamento são necessários pares de imagens e textos associados. O modelo treinado, por sua vez, sugere quais dados textuais possivelmente descreve uma nova imagem de teste. Ou seja, o modelo uma vez treinado tem como entrada uma imagem de teste e como saída dados textuais sugeridos.

O procedimento de treino envolve a definição de traços semânticos, uma abstração para um espaço latente que representa conjuntamente os dados textuais e visuais, discutida em detalhes ao longo deste documento. Os dados textuais considerados neste trabalho estão no formato de vetor de termos (ou *bag-of-words*), e assim, a saída do modelo é um vetor com os possíveis termos que descrevem uma imagem de teste.

Em linhas gerais, a arquitetura proposta envolve um conjunto de classificadores (Redes Neurais Profundas) específicas por aprender cada um dos traços semânticos específicos presentes nas imagens. A arquitetura também apresenta um conjunto de modelos generativos, treinados sobre os dados textuais para a identificação dos possíveis traços semânticos existente na base textual. Utilizando a capacidade generativa do conjunto de modelos generativos, os dados textuais são então reconstruídos no processo de predição do modelo a partir dos traços semânticos identificados pelo conjunto de classificadores.

Por fim, um protótipo para a arquitetura proposta foi implementado e avaliado em três grupos de experimentos. Como parte do protótipo implementado, foram usados *Stacked Auto-Encoders* (Auto Codificadores Empilhados) (SAE) baseados em redes pré-treinadas, além de uma proposta para uma regra de definição dos traços semânticos com base nos dados textuais disponíveis, passo necessário para o treinamento do modelo.

O primeiro grupo de experimentos teve como objetivo avaliar a capacidade de aprendizagem dos classificadores internos ao modelo. O segundo grupo de experimentos buscou avaliar a qualidade dos dados textuais estimados pelo modelo. A arquitetura proposta não tem como finalidade principal a classificação de imagens, a arquitetura proposta não se trata de um classificador, e sim uma técnica para estimar dados textuais associados para imagens de teste. Diante deste fato, um terceiro grupo de experimentação foi realizado para avaliar a qualidade dos dados textuais estimados pelo modelo proposto quando aplicados em uma tarefa de classificação de imagens, com o auxílio de um classificador regular presente na literatura.

Deste trabalho decorre a contribuição científica intitulada "*Transferring Knowledge From Texts to Images by Combining Deep Semantic Feature Descriptors*". Tendo o artigo publicado nos anais do evento "Conferência Internacional Unificada em Redes Neurais 2018" – "*The 2018 International Joint Conference on Neural Networks" (IJCNN 2018).*

1.4 Estrutura da Dissertação

Além deste capítulo introdutório, o restante desta dissertação está estruturada da seguinte forma:

O **Capítulo 2** apresenta conceitos essenciais na área da Aprendizagem Profunda. O capítulo apresenta inicialmente um breve histórico do campo das Redes Neurais Artificiais, partindo das primeiras contribuições da área até as contribuições mais recentes que consolidam a Aprendizagem Profunda. Por fim, conceitos básicos em Redes Neurais Profundas e Redes Neurais Convolucionais são apresentados.

O **Capítulo 3** apresenta as principais contribuições na área da Transferência de Conhecimento em Aprendizagem de Máquina. O capítulo apresenta um mapeamento da literatura acerca da problemática em uma abordagem *Top-Down*, apresentando as principais definições e formalizações da área, além dos conceitos mais consolidados. Por fim, uma seção trata especificamente da temática de Transferência de Conhecimento na área da Aprendizagem Profunda.

O **Capítulo 4** detalha a arquitetura da solução proposta para a realização da transferência de conhecimento dos domínios textuais para domínios visuais, através da aprendizagem de tópicos semânticos latentes. O capítulo apresenta a visão geral da arquitetura e procedimentos de treino e de predição, para então elaborar em mais detalhes a técnica proposta. O capítulo também descreve em detalhes o protótipo implementado para a técnica proposta, utilizado nos experimentos. Uma vez que a técnica proposta trata-se de uma arquitetura mais genérica, o capítulo também apresenta algumas particularidades do protótipo implementado, mas que não limitam o conceito mais amplo da arquitetura proposta.

O **Capítulo 5** apresenta em detalhes os experimentos realizados, bem como seus objetivos, resultados e uma discussão sobre os mesmos. Em linhas gerais, foram elaborados experimentos para avaliar os classificadores internos ao modelo proposto, avaliar a qualidade das informações geradas pelo modelo e a qualidade das informações geradas pelo modelo quando aplicadas para realizar uma tarefa de classificação de imagens. O capítulo também apresenta um estudo preliminar sobre a base de dados utilizada nos experimentos.

O **Capítulo 6** sumariza o trabalho realizado, apresentando as considerações finais sobre as contribuições realizadas e uma discussão final. Além disso, o capítulo aponta direções de trabalhos futuros e possíveis aspectos a serem evoluídos.

2 APRENDIZAGEM PROFUNDA

Este capítulo apresenta conceitos essenciais na área da Aprendizagem Profunda. O capítulo apresenta inicialmente um breve histórico do campo das Redes Neurais Artificiais, partindo das primeiras contribuições da área até as contribuições mais recentes que consolidam a Aprendizagem Profunda. Por fim, conceitos básicos em Redes Neurais Profundas e Redes Neurais Convolucionais são apresentados.

2.1 Breve Histórico das Redes Neurais Artificiais

A história das redes neurais artificiais tem início com a proposta do que seria a primeira "máquina" inspirada no cérebro humano, ou seja, o primeiro neurônio artificial. O trabalho proposto por McCulloch e Pitts (1943), apresenta uma definição para a ativação de um neurônio artificial, onde caso o somatório de entradas binárias superar um certo limiar, a saída desse neurônio deve ser ativada, se o somatório das ativações de entrada for inferior ao limiar o neurônio, por sua vez, deve manter-se inativo.

No entanto, o mecanismo proposto por McCulloch e Pitts não apresentava a capacidade de aprendizagem, ou seja, não havia uma definição de como informações poderiam ser armazenadas e recuperadas nos neurônios artificiais propostos. A contribuição de Hebb (1949) apresentou como neurônios poderiam aprender e armazenar informações baseando-se na troca de informações entre eles, a partir de ligações (sinapses) e ativações. De modo geral, a regra proposta por Hebb define que as ativações de um neurônio passa a se dar em função das ativações de outros neurônios ligados a este.

Em 1958, o trabalho apresentado por Rosenblatt (1958) colocou em prática as propostas de McCulloch e Pitts e Hebb em um mecanismo intitulado *Perceptron*, capaz de aprender e realizar reconhecimento de padrões, despertando grande interesse no campo das redes neurais artificiais. Em sequência, a regra de aprendizagem *Adaptive Linear Neuron* (Neurônio Adaptativo Linear) (ADALINE) foi apresentada por Widrow e Hoff (1960) como uma extensão para o *Perceptron*. A regra baseada no método dos mínimos quadrados ficou conhecida como regra *delta*, difundindo ainda mais a área das redes neurais artificiais.

O trabalho publicado por Minsky e Papert (1969) demonstrou uma limitação dos *Perceptrons*, estes poderiam apenas resolver problemas linearmente separáveis, levantando grandes discussões a respeito da área e desencadeando um desinteresse pela mesma, o que resultou em uma redução na quantidade de contribuições neste campo. Mais precisamente, o trabalho apontou a incapacidade de um *Perceptron* em solucionar um problema XOR – "OU-Exclusivo".

Mais tarde a área voltou a atrair o interesse de pesquisadores com o trabalho de Hopfield (1982), sobre a associatividade das redes neurais mais precisamente nas redes denominadas redes de Hopfield. Outra forte contribuição que despertou novamente o interesse na área foi o algoritmo *backpropagation* proposto na tese de doutorado de Werbos (1974). Essas duas contribuições introduziram uma abordagem de combinar redes neurais artificiais, possibilitando assim que um modelo de rede neural fosse capaz de solucionar problemas não linearmente separáveis, colocando a área em alta novamente. Para essa combinação de *Perceptrons* foi dado o nome de *Multilayer Perceptron* (MLP) – Perceptron Multicamadas (PM).

O *backpropagation* e sua utilização no treinamento de redes neurais artificiais ficou ainda mais difundido após o trabalho de Rumelhart, Hinton e Williams (1986), o que desencadeou muito mais interesse e contribuições na área. Mais tarde, o trabalho apresentado por LeCun et al. (1989) apresentou uma aplicação de mundo real bastante relevante, o reconhecimento de dígitos manuscritos para códigos postais, o que desencadeou ainda mais interesse pela área e provavelmente iniciou uma nova etapa na história das redes neurais. O trabalho *"Gradient-based learning applied to document recognition"* ("Aprendizagem Baseada em Gradiente Aplicada para Reconhecimento de Documento" de LeCun et al. (1998), apresenta uma arquitetura de rede neural com cinco camadas (mais camadas que a quantidade usualmente aplicada nas MLP), dentre elas camadas convolucionais, desempenhando uma atividade de reconhecimento de dígitos manuscritos. O trabalho representa uma nova linha de contribuições científicas na área de grande interesse, atualmente denominada Aprendizagem Profunda – *Deep Learninig*.

2.2 Tipos de Redes Neurais Profundas

Com o avanço da Aprendizagem Profunda, diversos tipos de redes neurais artificiais passaram a ser atribuídos à categoria de Redes Neurais Profundas. Dentre as categorias atuais, a mais notável é a *Convolutional Neural Network* (Rede Neural Convolucional) (CNN), apresentando ótimos resultados em tarefas de visão computacional. Mas além das Redes Neurais Convolucionais, existem outras arquiteturas relevantes que serão apresentadas ao longo dessa seção.

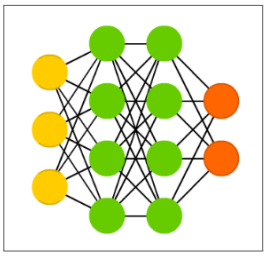
Após as contribuições de Hopfield (1982), Werbos (1974) e Rumelhart, Hinton e Williams (1986), as redes decorrentes da MLP ganharam bastante notoriedade, principalmente pela ideia de que adicionar camadas intermediárias (além da quantidade usualmente aplicada nas MLP) podem permitir que as redes denominadas Redes Neurais Profundas possam resolver problemas mais complexos, como pode ser observado no trabalho (LECUN et al., 1998). A seguir veremos algumas arquiteturas básicas de redes neurais. Para as ilustrações, foram adotadas representações dos neurônios conforme legenda apresentada na Figura 1.

Figura 1 – Legenda para as ilustrações das arquiteturas básicas de redes neurais

Fonte: adaptação de Veen (2016)

Redes *Feed Forward* são basicamente redes MLP, as quais apresentam uma camada de entrada, uma ou mais camadas escondidas e uma camada de saída. Os neurônios entre camadas vizinhas usualmente apresentam sinapses que os conectam completamente, caracterizando as chamadas Redes Totalmente Conectadas ou também chamadas de Redes Densas. O raciocínio das Redes *Feed Forward* caracteriza as camadas totalmente conectadas (ou densas), usualmente aplicadas no topo de outras arquiteturas para realizar a combinação de características e classificação, culminando na camada de saída. A Figura 2 ilustra a estrutura de uma Rede MLP.

Figura 2 – Ilustração da arquitetura de uma rede MLP

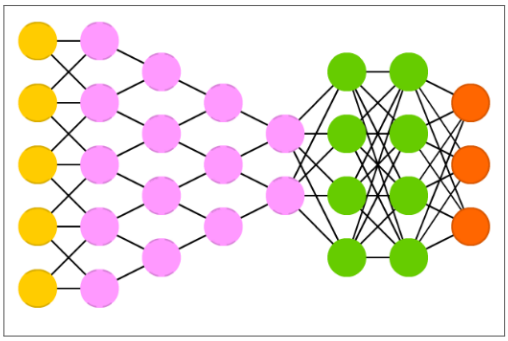


Fonte: adaptação de Veen (2016)

As Redes Neurais Convolucionais foram inicialmente representadas pela arquitetura LeNet e suas evoluções, como a LeNet-5 proposta por LeCun et al. (1998). Uma das principais características das CNN, ou apenas Redes Convolucionais, é a presença das camadas convolucionais e de amostragem (*pooling* ou *subsampling*) explicadas mais adiante nesta seção. As camadas convolucionais e de amostragem permitem tratar imagens de entrada de uma forma que preserva o aspecto espacial da imagem, ou seja, a vizinhança entre *pixels* é mantida e processada ao longo das camadas, o que demonstrou uma performance relevante em tarefas de visão computacional e processamento de imagens. Diversas arquiteturas de CNN foram propostas, evoluindo a LeNet, para realizar tarefas em bases de imagens cada vez mais complexas. Algumas das arquiteturas de CNN mais notáveis serão apresentadas na próxima seção. A Figura 3 ilustra a estrutura de uma CNN.

Os Auto Codificadores são variações não supervisionadas de redes neurais, onde o objetivo de um Auto Codificador é aprender a reconstruir os dados de entrada, com a

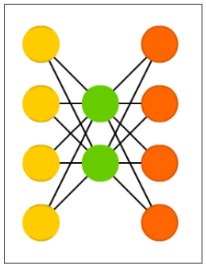
Figura 3 – Ilustração da arquitetura de uma Rede Neural Convolucional



Fonte: adaptação de Veen (2016)

limitação de representar a entrada em um espaço de características reduzido, na camada intermediária da rede. Utilizar Auto Codificadores empilhados origina as redes denominadas SAE. O processo de treinar um SAE pode ser definido em duas etapas, onde a primeira metade da rede aprende a representar a entrada em um espaço reduzido de características e a segunda metade da rede é responsável por reconstruir a entrada a partir da representação reduzida da mesma. Desse modo, a primeira metade da rede pode ser utilizada como uma Auto Codificador, aplicado para redução de dimensionalidade. A Figura 4 ilustra a estrutura de um SAE. Alternativamente, os SAE podem apresentar camadas convolucionais, originando os *Convolutional Autoencoder* (Auto Codificadores Convolucionais) (CAE), bastante utilizados como poderosos extratores de características para imagens, ilustrado na Figura 5.

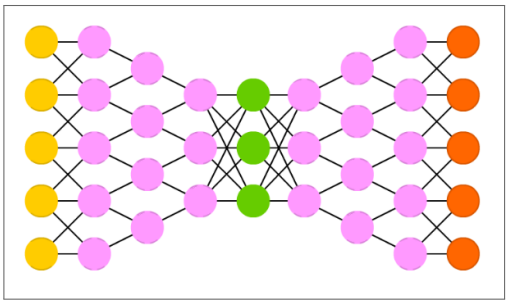
Figura 4 – Ilustração da arquitetura de um Auto Codificador



Fonte: adaptação de Veen (2016)

Outros tipos de redes neurais, igualmente relevantes para diferentes finalidades, também são classificadas como redes neurais profundas, muito embora essa classificação por vezes possa gerar discussões. Como exemplo de outros tipos de redes, temos as Redes Recorrentes, Redes Generativas Adversárias, *Deep Belief Network* (DBN), entre outras. Ademas, novas arquiteturas são propostas frequentemente, o que torna a área atualmente bastante dinâmica e em rápida evolução.

Figura 5 – Ilustração da arquitetura de um Auto Codificador Convolucional



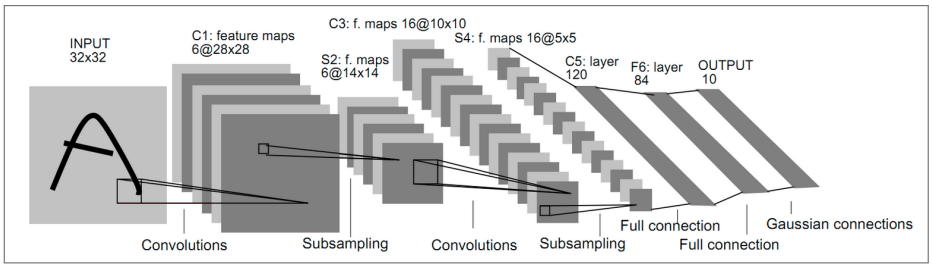
Fonte: adaptação de Veen (2016)

2.3 Arquiteturas de Redes Neurais Convolucionais

Como mencionado, as Redes Neurais Convolucionais (CNN) são uma das categorias das chamadas Redes Neurais Profundas. As principais arquiteturas de CNN apresentam uma camada de entrada para representar imagens, usualmente com largura, altura e canais de cores, camadas convolucionais no papel de um Auto Codificador para extrair características de alto nível e no final da rede camadas densas (ou totalmente conectadas) para combinar as características de alto nível e finalmente realizar alguma atividade específica, como classificação, detecção de objetos ou segmentação. Variações sem as camadas densas finais também foram propostas, originando as denominadas Redes Totalmente Convolucionais.

A primeira arquitetura notável já foi mencionada neste capítulo, trata-se da LeNet-5, proposta por LeCun et al., projetada para reconhecer dígitos manuscritos em códigos postais. A arquitetura apresenta cinco camadas, das quais as duas primeiras são camadas convolucionais e as três últimas são camadas totalmente conectadas para realizar a combinação das características e uma classificação final na camada de saída. A Figura 6 ilustra a arquitetura da LeNet-5.

Figura 6 – Ilustração da arquitetura da LeNet-5

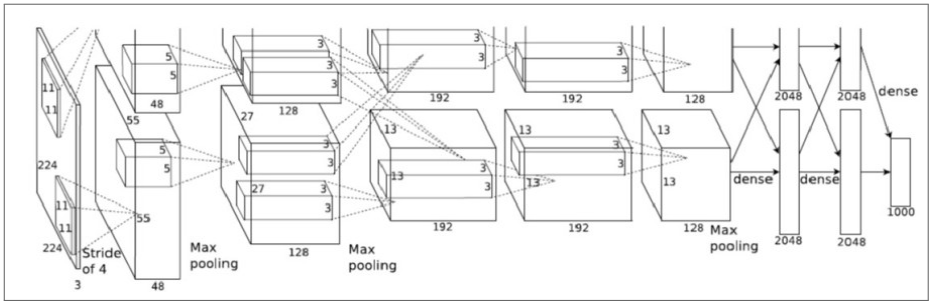


Fonte: LeCun et al. (1998)

O trabalho apresentado por Krizhevsky, Sutskever e Hinton (2012) apresenta a *AlexNet*, uma rede com 8 camadas, das quais as 5 primeiras são camadas convolucionais e as 3 últimas são camadas totalmente conectadas. A Figura 7 ilustra a arquitetura da *Alex-*

Net. O trabalho também apresenta uma alternativa de treinamento em paralelo da rede, dividindo o processamento em dois *hardwares* específicos para processamentos gráficos.

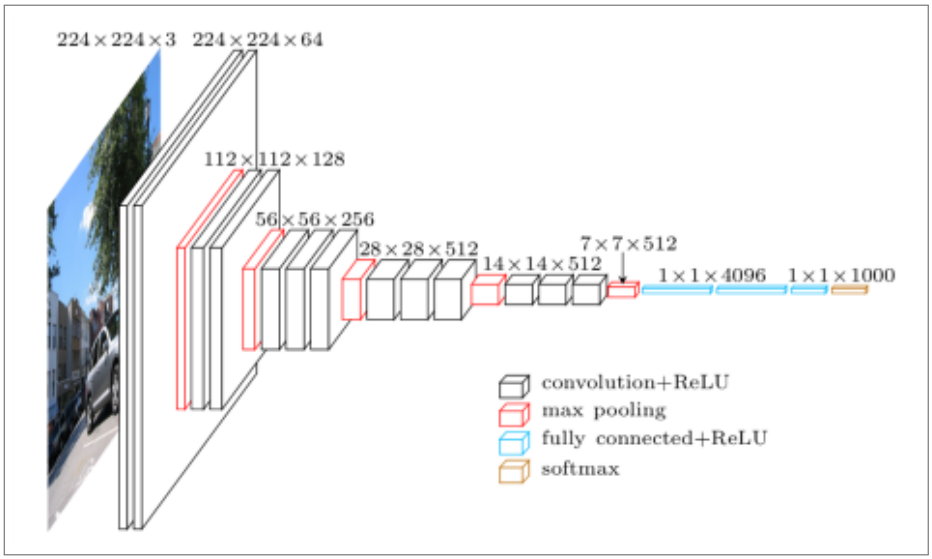
Figura 7 – Ilustração da arquitetura da *AlexNet*



Fonte: Krizhevsky, Sutskever e Hinton (2012)

A *Visual Geometry Group* (VGG), é uma família de arquiteturas de Redes Neurais Convolucionais, com variações de 11, 13, 16 e 19 camadas, das quais as três ultimas camadas são fixas do tipo totalmente conectadas, com 4096, 4096 e 1000 neurônios, respectivamente, e as primeiras são camadas convolucionais. A VGG foi a arquitetura vencedora do *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC) do ano de 2014, sendo bastante referenciada e utilizada desde então. O trabalho original foi publicado no trabalho de Simonyan e Zisserman (2014). A Figura 8 ilustra a arquitetura da VGG-16, a variação com 16 camadas da VGG.

Figura 8 – Ilustração da arquitetura da VGG-16



Fonte: Blier (2016)

Outras arquiteturas de Redes Convolucionais continuam sendo propostas com relativa velocidade, com o objetivo de reduzir o tamanho em memória ocupado pelos modelos treinados, treinar redes mais precisas, além de projetar arquiteturas para aplicações em visão computacional.

Como exemplo podemos mencionar ainda a *ResNet* e suas evoluções (HE et al., 2016), a *Inception* (SZEGEDY et al., 2015), a *MobileNet* (HOWARD et al., 2017), a *SqueezeNet* (IANDOLA et al., 2016), entre outras.

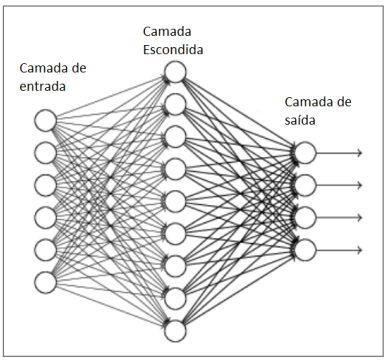
2.4 Elementos das Redes Neurais Convolucionais

Nesta seção são apresentados alguns conceitos básicos sobre os principais elementos presentes nas principais arquiteturas de Redes Neurais Convolucionais, necessários para o entendimento deste trabalho.

2.4.1 Tipos de camadas

Camadas totalmente conectadas, ou **camadas densas**, são camadas de neurônios onde todos os neurônios da camada apresentam sinapses (conexões) com todos os neurônios da camada de entrada (em relação à própria camada, e não a camada de entrada da rede). Camadas densas surgiram inicialmente com os conceitos das primeiras MLPs, e ainda hoje são bastante aplicadas em arquiteturas de Redes Neurais Profundas como camadas finais com a finalidade de combinar características de alto nível, resultantes de camadas anteriores. Nas camadas totalmente conectadas, cada sinapse apresenta um peso a ser aprendido pelo otimizador, tornando assim as camadas densas responsáveis por grande parte do tamanho final do modelo treinado. A Figura 9 ilustra uma camada densa, no papel da camada escondida de uma MLP clássica.

Figura 9 – Ilustração de uma camada escondida, totalmente conectada ou densa



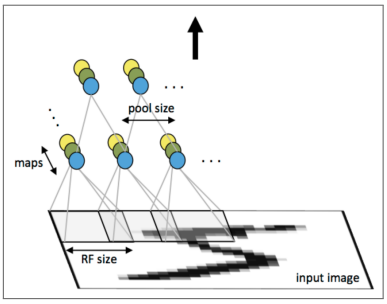
Fonte: Wanderley (2018)

As **camadas convolucionais** caracterizam fortemente as Redes Neurais Convolucionais, nessas camadas, os neurônios desempenham o papel de filtros receptivos que realizam operações de reconhecimento de padrões em intervalos, seja no espaço unidimensional ou bidimensional, sendo o caso da convolução bidimensional mais comum para o tratamento de imagens, *i.e.*, camadas convolucionais 2-D. Similar a uma operação de convolução, os

neurônios realizam uma varredura na imagem de entrada, dentro de uma janela te tamanho definido, e caso um padrão seja identificado, a camada propaga a ativação de seus neurônios. Camadas convolucionais apresentam dois grandes benefícios, primeiro que uma camada convolucional é capaz de avaliar uma camada de entrada que representa uma imagem com uma quantidade bem menor de neurônios, os os mesmos pesos são usados para varrer espacialmente a imagem. E um segundo benefício é que as camadas convolucionais respeitam os aspectos espaciais (vizinhança entre *pixels*) das imagens de entrada, sem a necessidade de linearizar a imagem original.

Usualmente, após camadas convolucionais são utilizadas camadas de sub-amostragem ou *pooling*, essas camadas não apresentam neurônios com pesos ajustados durante o treinamento, mas são responsáveis por reduzir o tamanho da camada de entrada, realizando operações simples (máximo, média, entre outras) entre um grupo de *pixels* vizinhos. A combinação de camadas convolucionais e camadas de *pooling*, além de uma ativação não linear mencionada adiante, definem uma terminologia mais abstrata para a definição das camadas convolucionais, de acordo com (GOODFELLOW et al., 2016). A Figura 10 ilustra as camadas convolucionais e de *pooling*.

Figura 10 – Ilustração de uma camada convolucional



Fonte: Ng et al. (2016)

2.4.2 Funções de Ativação

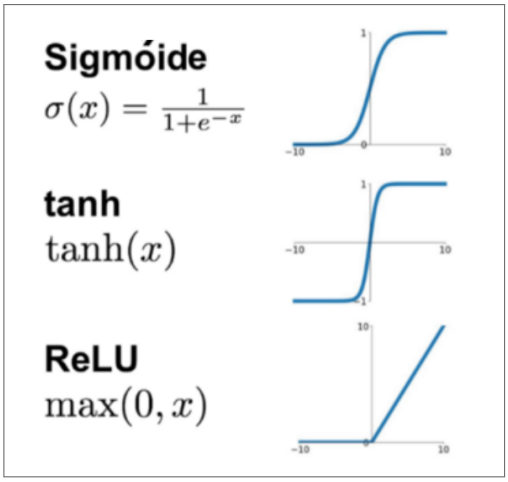
Os neurônios de uma camada apresentam funções de ativação, para determinar se um impulso deva ser propagado para as camadas seguintes, com base nos impulsos recebidos pelo neurônio em questão.

A primeira função de ativação a ser mencionada é a função **sigmóide**, uma função bastante utilizada nas primeiras MLPs, apresenta uma variação no intervalo [0,1], não é centrada no zero e apresenta uma operação exponencial em sua definição. O principal problema da sigmóide é que esta satura o gradiente para valores extremos, uma vez que a derivada da função se aproxima do zero para valores muito elevados (e igualmente valores negativos muito elevados).

Outra função de ativação bastante utilizada é a **tangente hiperbólica (tanh)**, a função é centrada no zero, e varia no intervalo [-1,1]. A função ainda pode apresentar o problema de saturação da sigmóide.

***Rectified Linear Unit* (ReLU)** é a função de ativação mais comumente usada, devido à sua simplicidade de diferenciação e não é computacionalmente cara. A função também contorna o problema da saturação apresentado pela tanh e a sigmóide. A ReLU é bastante utilizada como função de ativação em camadas intermediárias em uma rede profunda. A Figura 11 apresenta as funções mencionadas e seus gráficos.

Figura 11 – Quadro com diferentes funções de ativação e seus gráficos



Fonte: Wanderley (2018)

Uma outra função baste utilizada para camadas de classificação é a ***SoftMax***, com a característica de apresentar as probabilidades para cada classe envolvida, ou seja, os valores da função variam no intervalo [0,1] e a soma dos valores é igual a 1. A ativação *SoftMax* é usualmente aplicada em camadas de saída, em arquiteturas de redes neurais desenhadas para atividades de classificação, dada a interpretação probabilística gerada pela ativação.

3 TRANSFERÊNCIA DE CONHECIMENTO

Aplicações reais de Aprendizagem de Máquina exigem dados necessários para treinamento de modelos e dados para teste destes modelos, para que então estes sejam colocados em produção, ou seja, sejam expostos aos dados reais do problema. Em termos práticos, os dados utilizados para treinamento e teste dos modelos precisam estar disponíveis de antemão e devidamente rotulados, mas eventualmente quando os modelos são expostos para os dados da aplicação alvo, os mesmos desempenhos observados no domínio do treinamento podem não ser observados. Isso indica que nem sempre os dados do domínio onde um modelo foi treinado seguem a mesma distribuição dos dados do domínio onde o modelo irá operar, isso se deve ao fato de que os dados utilizados no treinamento são provavelmente uma amostra pouco representativa do espaço real da aplicação ou que os dados são provenientes de um domínio diferente do qual pretende-se utilizar o modelo. O fato é que, em diversas ocasiões precisa-se treinar um modelo para um determinado domínio onde existem poucos dados disponíveis para treino, mas existem dados com maior disponibilidade e outros modelos treinados em domínios similares que podem auxiliar o treinamento e o desempenho de um novo modelo em uma nova, porém similar, tarefa.

3.1 Transferência de Aprendizagem

O cenário mencionado descreve uma problemática que busca ser resolvida por técnicas de Transferência de Aprendizagem (ou *Transfer Learning*), no campo da Aprendizagem de Máquina, que tem como objetivo utilizar conhecimento existente em dados e modelos de um domínio denominado fonte, para melhorar o desempenho de um modelo treinado em um domínio alvo onde pouco ou nenhum conhecimento prévio está disponível. Estas características podem ser observadas em diversas áreas de aplicação, como processamento de linguagem natural, análise de sentimento, visão computacional, biologia, setor financeiro, entre outros.

3.1.1 Formalização

As principais definições do contexto de Transferência de Aprendizagem (ou *Transfer Learning*) são formalizado no trabalho *"A survey on transfer learning"* de Pan, Yang et al. (2010), onde são apresentadas três principais definições: Domínio (*Domain*), Tarefa (*Task*) e Transferência de Aprendizagem (*Transfer Learning*), formalizadas da seguinte forma:

Domínio: $D = \{\chi, P(X)\}$, onde χ denota o espaço de características e a distribuição de probabilidade marginal P(X), onde $X = \{x_1, x_2, ..., x_n\} \in \chi$.

Tarefa: $T = \{Y, f(.)\}$, onde Y denota o espaço de rótulos $Y = \{y_1, y_2, ..., y_m\}$ e uma função preditiva objetivo $f(.)$ não observada, aproximada a partir das instâncias de exemplos e respectivos rótulos, *i.e.*, os pares $\{x^i, y^i\}$.

Transferência de Aprendizagem: Dado um domínio fonte (*Source Domain*) D^s com uma tarefa fonte T^s relacionada e um domínio alvo (*Target Domain*) D^t com uma tarefa T^t alvo relacionada, sabendo que $D^s \neq D^t$ ou $f^s(.) \neq f^t(.)$, a Transferência de Aprendizagem se dá quando o conhecimento presente em D^s e T^s são utilizados para melhorar o treinamento e eficiência da função objetivo $f^t(.)$ relacionada com o domínio D^t.

A definição $D^s \neq D^t$ implica que $\chi^s \neq \chi^t$ ou $P^s(X) \neq P^t(X)$ e ainda, $T^s \neq T^t$ sugere que $Y^s \neq Y^t$ ou $T^s \neq T^t$. Observe que o contexto de Transferência de Aprendizagem se torna um problema tradicional de Aprendizagem de Máquina quando $D^s = D^t$ e $T^s = T^t$.

3.1.2 Categorias de Transferência de Aprendizagem

Com base na definição apresentado por Pan, Yang et al. (2010), três principais categorias de Transferência de Aprendizagem podem ser observadas:

Transferência Indutiva: *Inductive Transfer Learning*, caracteriza os problemas onde a tarefa e o domínio fonte são diferentes da tarefa e domínio alvo, ou seja, $T^s \neq T^t$ e $D^s \neq D^t$. Na Transferência Indutiva temos os casos onde as tarefas e os domínios diferem, mas de certo modo estão relacionados e podem ser mutuamente úteis, por exemplo, problemas de reconhecimento de dígitos e problemas de reconhecimento de letras.

Transferência Transdutiva: *Transductive Transfer Learning*, define os problemas onde o domínio fonte é diferente do domínio alvo mas as tarefas de aprendizagem são as mesmas em ambos os domínios, $T^s = T^t$ e $D^s \neq D^t$. Neste tipo de transferência, a tarefa é compatível, mas os domínios diferem, como por exemplo a classificação de imagens de animais e a classificação áudios de animais, quando o conjunto de animais é o mesmo entre os domínios.

Transferência Não-Supervisionada: *Unsupervised Transfer Learning*, similar à transferência indutiva, mas voltado para os problemas não supervisionados, como problemas de clusterização, redução de dimensionalidade ou até mesmo os casos onde todos ou parte dos dados do domínio fonte não estão rotulados, $T^s \neq T^t$ e $D^s \neq D^t$. Por exemplo, utilizar o conhecimento aprendido na redução de características de imagens de mundo real para imagens de documentos.

3.1.3 Domínio Fonte Simples e Domínio Fonte Múltiplo

Ainda com base na definição apresentada por Pan, Yang et al. (2010), um caso alternativo bastante observado em aplicações reais pode ser definido: Domínio Fonte Múltiplo (*Multiple Source*). Até o momento, a definição apresentada descreve o caso de Domínio

Fonte Simples, onde os dados do domínio fonte são provenientes de uma mesmo contexto, ou seja, os dados X seguem a mesma distribuição $P(X)$ (*Single Source*). Já no caso de um domínio fonte múltiplo, os dados do domínio fonte são provenientes de mais de um contexto, apresentando assim mais de uma distribuição marginal de probabilidade. Em linhas gerais, a formalização de um domínio fonte múltiplo com d diferentes domínios se torna a união destes diferentes domínios, ou seja, $D^s = \{D_1^s, D_2^s, ..., D_d^s\} = \{\{\chi_1^s, P_1^s(X_1^s)\}, \{\chi_2^s, P_2^s(X_2^s)\}, ..., \{\chi_d^s, P_d^s(X_d^s)\}\}$

3.2 Mapeamento da Literatura

Nos últimos anos diversas contribuições foram publicadas abordando a temática de Transferência de Aprendizagem, muitos trabalhos seguem a mesma temática sob nomes similares que convergem para a mesma definição de problema, como "adaptação de domínio", "reuso de modelo", "transferência de modelo", "*Concept Drift*", entre outros. As contribuições variam desde soluções para contextos e problemas específicos até algoritmos mais genéricos, que podem ser aplicados em uma variedade maior de contextos, mas a grande maioria aborda problemas específicos com soluções específicas, o que dificulta a aplicação da mesma técnica em outros contextos.

O trabalho *"Transfer Learning Using Computational Intelligence: A Survey"* de Lu et al. (2015), aponta que, na literatura, a categoria de Transferência de Aprendizagem Transdutiva (*Transductive Transfer Learning*) bem como os termos *"Domain Adaptation", "Covariate Shift", "Sample Selection Bias", "Transfer Learning", "Multi-Task Learning", "Robust Learning" e "Concept Drift"* são todos termos comumente aplicados para descrever o mesmo cenário de aplicação. O seja, o problema particular tratado pela Transferência de Aprendizagem onde as tarefas de aprendizagem são as mesmas mas o domínio fonte é diferente do domínio alvo ($T^s = T^t, D^s \neq D^t$) se torna um problema relevante e bastante abordado na literatura.

A primeira principal contribuição mencionada no trabalho de Lu et al. (2015) é o agrupamento dos trabalhos em quatro principais grupos de aplicação para problemas relacionados com adaptação de domínio. O primeiro grupo consiste em atribuir pesos para instâncias de treinamento do domínio fonte para adequar este ao domínio alvo via técnicas de "Covariate Shift" (LU et al., 2015). O segundo grupo refere métodos de "auto-rotulação", onde instâncias não rotuladas do domínio alvo são adicionadas no processo de treinamento, tendo estes rótulos inicializados e ajustados ao longo do processo, de acordo com os autores, apresenta forte relação com o algoritmo de *Expectation–Maximization* (EM) (DEMPSTER; LAIRD; RUBIN, 1977). O terceiro grupo inclui métodos de representação do espaço de características, que buscam encontrar uma representação dos espaços de características fonte e alvo (χ^s, χ^t) de modo a ficarem similar. E o quarto grupo representa os métodos baseados em clusterização, onde o agrupamento das instâncias dos domínios

possibilita identificar o rótulo de instancias alvo não rotuladas.

Ainda no *survey* de Lu et al. (2015), os principais tipos de abordagens para o problema de Transferência de Aprendizagem foram as técnicas de redes neurais (incluindo *Deep Neural Networks* e *Multiple Task Neural Networks*), técnicas estatísticas – Bayes (incluindo *Naive Bayes*, Redes Bayesianas e Modelo Bayesiano Hierárquico) e técnicas envolvendo sistemas difusos e algoritmos genéticos. Foram apontados como os principais domínios de aplicações reais as áreas: Processamento Natural de Linguagem, Visão Computacional, Biologia, Finança e Gestão de Negócios. No total foram referenciados 30 trabalhos, onde a técnica mais explorada foi Rede Neural (com 17 trabalhos) e o domínio mais investigado foi visão computacional e processamento de imagens (com 12 trabalhos citados).

O trabalho "A Survey of Transfer Learning" de Weiss, Khoshgoftaar e Wang (2016) no *"Journal of Big Data"*, aponta que desde a publicação do trabalho de Pan, Yang et al. (2010), mais de 700 trabalhos foram escritos, apresentando avanços e melhorias em Transferência de Aprendizagem. Os trabalhos investigados neste 5 anos de contribuições concentram-se em 2 principais grupos de Transferência de Aprendizagem: **Transferência Homogênea de Aprendizagem** (quando os espaços de características entre os domínios fonte e alvo são equivalentes) apontando 18 algoritmos (CP-MDA, 2SW-MDA, FAM, DTMKTL, JDA, ARTL, TCA, SFA, SDA, GFK, DCP, TCNN, MMKT, DSM, MsTrAdaBoost, TaskTrAdaBoost, RAP e SSFE) e **Transferência Heterogênea de Aprendizagem** (quando os espaços de características entre os domínios fonte e alvo são diferentes) mencionando 12 algoritmos (CLSCL, HeMap, DAMA, HTLIC, TTI, HFA, SHFA, ARC-t, MOMAP, SHFR, HDP e HTL).

O estudo de Weiss, Khoshgoftaar e Wang (2016) ainda aponta a relevância do tema de Transferência Negativa, quando o domínio fonte prejudica a performance da tarefa alvo no domínio alvo, uma vulnerabilidade que pode estar presente nos algoritmos e pouco investigada nos trabalhos. Segundo os autores, a maioria das abordagens de Transferência de Aprendizagem busca ajustar a distribuição dos dados para diminuir a diferença entre os domínios. O *survey* apresenta diversas direções de pesquisa, como investigar a diferença entre abordagens de transferências de um estágio (quando o treinamento do modelo final/alvo é feito simultaneamente com a adaptação de domínio, uma tendência nos trabalhos mais recentes) e abordagens mais tradicionais de dois estágio (onde primeiramente é feita uma adaptação de domínio para então treinar um modelo). Outra contribuição relevante apresentada no trabalho é o desenvolvimento de um repositório único e padronizado com os códigos dos algoritmos, para possibilitar comparações mais justas e precisas entre as técnicas.

3.3 Transferência Heterogênea de Aprendizagem

Conforme discutido, a Transferência Heterogênea de Aprendizagem trata de problemas onde os domínios fonte e domínios alvo são de natureza diferentes, multimodais, mais precisamente, definem espaços de características distintos. Nesta seção discutiremos técnicas que tratam de uma manifestação particular desta problemática, onde os domínios em questão são os domínios textuais e visuais.

3.3.1 Transferência de Aprendizagem entre Domínios Textuais e Visuais

As técnicas apresentadas a seguir apresentam soluções que visam possibilitar a aprendizagem de informações presentes entre domínios visuais e textuais. As técnicas apresentadas possuem o objetivo de transferir conhecimento entre os domínios mencionados, utilizado como dados para treino pares de dados visuais e textuais associados e desempenham tarefas de classificação.

3.3.2 HTL

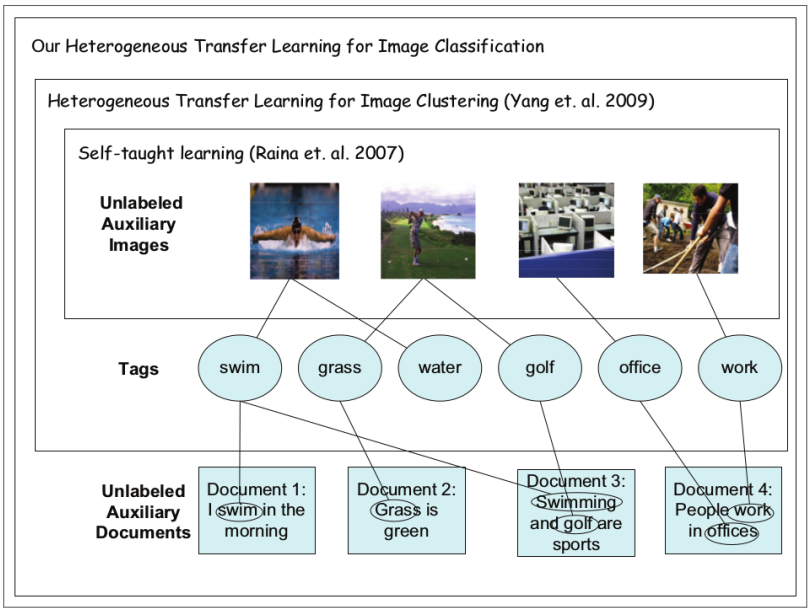
O trabalho apresentado por Zhu et al. (2011) nomeado ***Heterogeneous Transfer Learning for Image Classification*** **(Transferência de Aprendizagem Heterogênea para Classificação de Imagens) (HTL)**, apresenta uma técnica que utiliza três fonte de dados, são elas imagens auxiliares, *tags* das imagens auxiliares e textos auxiliares e define um espaço de características compatível com as três fontes de dados, chamado de espaço semântico latente.

De acordo com os autores, o diferencial do trabalho em relação aos trabalhos anteriores apresentados está na capacidade de aproveitar o conhecimento semântico presente na abundante quantidade de textos auxiliares, possibilitando a criação de um espaço de características comum com um significado semântico entre as três fontes de dados.

A técnica apresentada baseia-se em fatoração de matriz, onde a relação entre as imagens marcadas e as *tags* relacionadas é representada em um espaço semântico latente. Um conjunto de imagens marcadas (co-ocorrências de imagens e *tags* de texto) é usado para aprender uma visão semântica para cada representação de baixo nível de imagem, ou seja, treinar o modelo por uma fatoração de matriz. Uma nova instância de imagem alvo é mapeada para a representação semântica, então um classificador padrão pode ser construído na nova representação de imagem, para uma tarefa de classificação. A Figura 12 ilustra a arquitetura HTL.

Os experimentos avaliaram o desempenho do modelo proposto em contraste com outras técnicas relacionadas. Nos experimentos foram consideradas como imagens alvo a base de imagens Caltech-256 (GRIFFIN; HOLUB; PERONA, 2007), as imagens anotadas auxiliares e os textos auxiliares foram coletadas do portal de compartilhamento de imagens *Flickr*

Figura 12 – Ilustração da técnica HTL



Fonte: Zhu et al. (2011)

(FLICKR; LUDICORP; YAHOO!, 2004) e do engenho de busca de imagens do *Google*. A base de imagens Caltech-256, por combinação, define 171 pares de classes para problemas binários de classificação de imagens. Os resultados do modelo proposto, em comparação com outras técnicas relacionadas, se mostraram superiores em termos do resultado médio de acurácia dentre todas as 171 tarefas de classificação.

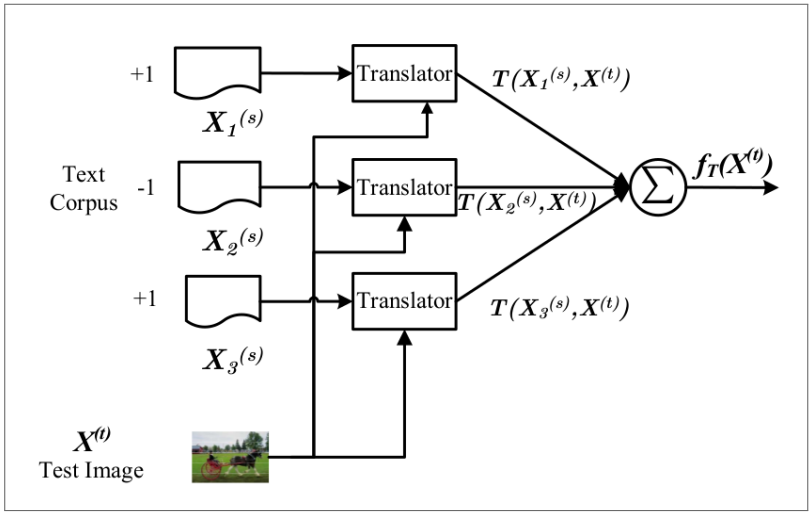
3.3.3 TTI

O trabalho apresentado por Qi, Aggarwal e Huang (2011) apresenta o método ***Translator from Texts to Images* (Tradutor de Textos para Imagens) (TTI)** para transferir conhecimento entre um domínio textual e domínio visual para tarefas de classificação de imagens. Segundo os autores, um dos objetivos do trabalho é desenvolver um modelo matemático para efetivamente reconhecer a relação entre características visuais e textos, e dessa forma, transferir indiretamente conhecimento semântico através de transformações de características.

A técnica baseia-se em um tradutor aprendido das co-ocorrências de textos e imagens. O tradutor faz a ponte entre os domínios de texto e imagem em um *espaço de tópicos* latente e é otimizado pelo método "gradiente proximal" proposto. O tradutor propaga rótulos semânticos do corpo de texto para as imagens, a fim de melhorar uma tarefa de classificação de imagens, conforme ilustrado na Figura 13 (QI; AGGARWAL; HUANG, 2011). A técnica proposta apresenta tradutores (funções) para realizar a transformação das características visuais de entrada pra instâncias do *corpus* textual, já a saída do modelo é realizada por uma função discriminante final, responsável por determinar o rótulo

final de predição. Para desempenhar a função objetivo de tradução entre os domínios, o trabalho propõe uma técnica de fatoração de matrizes denominada “Otimização Baseada em Gradiente Próximo".

Figura 13 – Ilustração da técnica TTI



Fonte: Qi, Aggarwal e Huang (2011)

Para apresentar a eficiência do modelo proposto, em contraste com outras técnicas relacionadas (inclusive a HTL), o trabalho conduz experimentos de classificação considerando uma base de dados de imagens e textos associados extraídos do portal de compartilhamento de imagens *Flickr* (FLICKR; LUDICORP; YAHOO!, 2004) e do engenho de busca de imagens do *Google*, dentre dez diferentes categorias. Os experimentos consideraram problemas de classificação binários, um para cada uma das dez categorias coletadas para compor a base de dados. Os resultados se mostraram superiores em termos da acurácia e taxa de erro de classificação das imagens para cada categoria, em comparação com as linhas de base consideradas no trabalho.

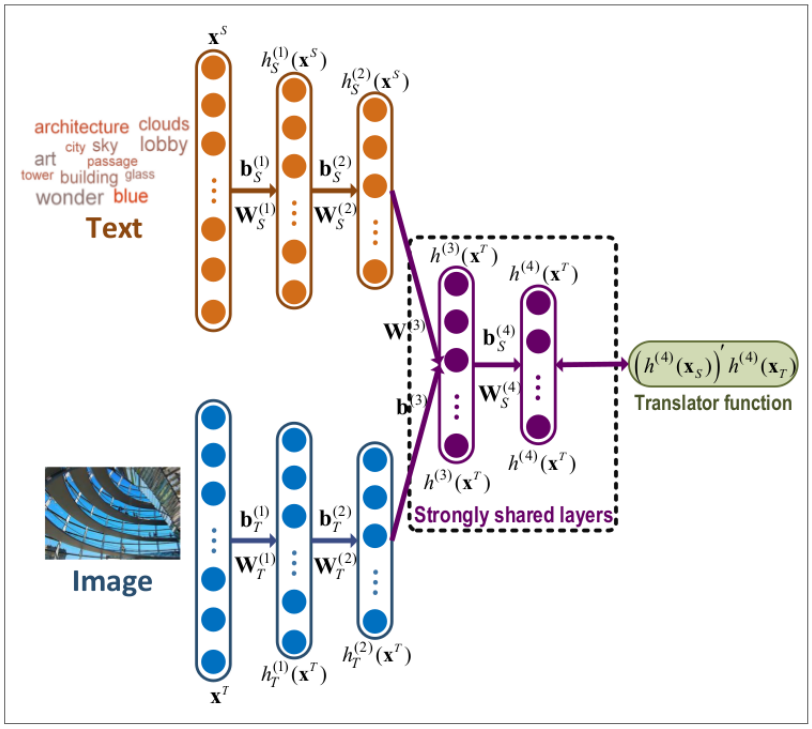
3.3.4 DTN

Os trabalhos apresentados por Zhang et al. (2015) e Shu et al. (2015) elaboram sobre as ***Deep Transfer Networks*** **(Redes de Transferência Profunda) (DTN)**, arquiteturas de redes neurais profundas para desempenhar a tarefa de transferência heterogênea de conhecimento entre os domínios visuais e textuais. Em comparação com as duas técnicas anteriores (baseadas em fatoração de matriz) esta técnica se diferencia pela sua abordagem baseada em redes neurais profundas.

A estrutura da rede neural proposta baseia-se em SAE empilhados, representada na Figura 14. A base da arquitetura (as primeiras camadas da rede neural, desde a camada de entrada) é um par de SAE, cada um treinado em cada domínio específico – textual e visual. As saídas das SAEs da base da rede são então combinadas e compõem a entrada do topo

da arquitetura, outra SAE por sua vez. O SAE do topo tem suas camadas compartilhadas, isto é, os pesos dos neurônios são compartilhados entre os dois primeiros SAEs, criando deste modo uma estrutura que codifica as duas entradas de domínio diferentes em uma representação em um espaço de características compartilhado(ZHANG et al., 2015; SHU et al., 2015).

Figura 14 – Ilustração da arquitetura de DTN



Fonte: Shu et al. (2015)

Os experimentos realizados para avaliar o desempenho das DTN em comparação com outras técnicas relacionadas (inclusive a HTL e TTI) utilizaram a base de dados NUS-WIDE (CHUA et al., 2009) que consiste em imagens e dados textuais (*tags*) associados, coletados do portal de compartilhamento de imagens *Flickr* (FLICKR; LUDICORP; YAHOO!, 2004). Da base de dados foram consideradas dez classes diferentes, definindo-se dez problemas binários de classificação individuais, um para cada classe, onde os exemplos positivos são aqueles que pertencem à classe em questão e os exemplos negativos são instâncias aleatórias de qualquer outra classe, dentre as dez selecionadas. Os resultados reportados pelos autores apontam um melhor desempenho em acurácia em relação aos trabalhos relacionados considerados, mesmo para experimentos com um conjunto reduzido de instâncias para treino do modelo.

3.4 Transferência de Aprendizagem em Aprendizagem Profunda

Muitos pesquisadores investigaram a reutilização de características aprendidas em redes neurais profundas e comprovaram o sucesso da Transferência de Aprendizagem. No trabalho "*Learning and transferring mid-level image representations using convolutional neural networks*" ("Aprendendo e Transferindo Representações de Nível Médio de Imagens Usando Redes Neurais Convolucionais") apresentado por Oquab et al. (2014), os autores aplicaram uma *AlexNet* (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) treinada no base de imagens ImageNet (RUSSAKOVSKY et al., 2015) para obter resultados de última geração nas tarefas de classificação e reconhecimento de ações do PASCAL VOC 2012 (EVERINGHAM et al., 2012; EVERINGHAM et al., 2010), demonstrando o potencial da Transferência de Aprendizagem de características de nível médio. Uma abordagem comum para a Transferência de Aprendizagem é realizar o treinamento de um classificador linear na parte superior da última camada de uma rede neural convolucional.

O estudo apresentado por Zeiler e Fergus (2014) reforçaram os benefícios da abordagem de pré-treinar uma CNN na base ImageNet (RUSSAKOVSKY et al., 2015), e então treinar um classificador linear para realizar uma tarefa nas bases PASCAL VOC 2012 (EVERINGHAM et al., 2012), Caltech-101 (FEI-FEI; FERGUS; PERONA, 2007) e Caltech-256 (GRIFFIN; HOLUB; PERONA, 2007). Variando o tamanho do conjunto de dados alvo e a camada a partir da qual o classificador é treinado, os autores observaram que o modelo generaliza bem para o Caltech-101 e o Caltech-256, mas não tanto para o PASCAL. Da mesma forma, bons resultados foram encontrados por Razavian et al. (2014) usando a mesma abordagem. Os autores combinaram um classificador *Support Vector Machine* (Máquina de Vetores de Suporte) (SVM) com uma CNN e pré-treinaram a rede na base ImageNet. Eles então usaram os conjuntos de dados Pascal VOC e MIT-67 *Indoor Scenes* (QUATTONI; TORRALBA, 2009) como tarefas alvo.

Em (DONAHUE et al., 2014), os autores examinaram como as características são transferidas para diferentes tarefas em diferentes domínios alvo e investigaram em que camada da rede neural a transferência é melhor. Os autores inicialmente treinaram uma *AlexNet* na base de imagens ImageNet e avaliaram os pesos aprendidos em uma tarefa básica de reconhecimento de objetos usando o conjunto de imagens da base Caltech-101. Depois disso, eles testaram a adaptação do domínio, onde apenas um pequeno conjunto de amostras está disponível, aplicando a base de imagens *Office* (SAENKO et al., 2010). Em terceiro lugar, eles avaliaram o desempenho do modelo em um conjunto de dados mais refinado, usando o conjunto de dados de aves Caltech-UCSD (WELINDER et al., 2010). Como as imagens neste conjunto de dados são muito semelhantes entre si, essa tarefa de classificação é considerada difícil. Por fim, os autores testaram o modelo no banco de dados SUN-397 *Large-Scale Scene Recognition* (XIAO et al., 2010). Nesse caso, a tarefa de destino é muito diferente da tarefa de origem. O conjunto de dados SUN-397 é usado para classificar categorias cênicas. Em todos os experimentos, as pontuações de comparação

(*benchmarks*) foram melhoradas, demonstrando que as características visuais aprendidas da base de imagem ImageNet fornecem propriedades generalizáveis.

No trabalho "*How transferable are features in deep neural networks?*" ("Quão Transferível São as Características em Redes Neurais Profundas?") de Yosinski et al. (2014), os autores analisaram o desempenho das características visuais transferidas ao longo das camadas de uma CNN *AlexNet*. Para este fim, os autores treinaram diferentes modelos em partições da base de imagens ImageNet, dividindo em duas partes o total de classes da base, denominadas baseA e baseB. Após o treinamento dos modelos básicos, os pesos aprendidos de cada camada foram transferidos para novas redes a serem treinadas na mesma base de dados ou em outras partições da base. Neste estudo, os autores observaram que as três primeiras camadas da rede representam filtros genéricos e reutilizáveis semelhantes aos filtros Gabor e *blobs* de cores e podem ser transferidos para melhorar o desempenho do classificador, mesmo se o conjunto de imagens alvo for de um domínio diferente. As camadas mais profundas, quaisquer que sejam, contêm informações mais específicas e especializadas para a tarefa inicial, e assim, transferi-las diminui o desempenho do classificador, mesmo ao aplicar um ajuste fino nas camadas transferidas.

Expandindo o trabalho de Yosinski et al. (2014), o trabalho apresentado por Soekhoe, Putten e Plaat (2016), investigou o efeito do tamanho do conjunto de dados de destino na Transferência de Aprendizagem, também usando um modelo *AlexNet*. A hipótese avaliada pelos autores é que uma abordagem de Transferência de Aprendizagem pelo "congelamento" das primeiras camadas resulta em uma tarefa de classificação mais eficiente, nos casos onde o tamanho do conjunto alvo é menor, em contraste com conjuntos de dados maiores, nos quais a atualização de todas as camadas proporcionará melhores resultados. Seguindo uma abordagem semelhante ao estudo de Yosinski et al., os autores validaram sua ideia em diferentes domínios de imagens, alterando o tamanho do conjunto de dados alvo ao testar os pesos transferidos de cada camada. Eles concluíram que, ao transferir as primeiras duas ou três camadas para conjuntos de dados alvo, com menos de mil instâncias por classe, o "congelamento" dos pesos aumenta o desempenho em relação aos métodos tradicionais. Ou seja, carregar os pesos da rede, mas não treinar novamente no domínio alvo resulta em uma maior acurácia de classificação. Embora extensos, esses estudos não avaliaram o efeito da complexidade e do tamanho dos conjuntos de dados do domínio alvo e fonte, durante o processo de transferência. Embora várias métricas sejam usadas na engenharia, a definição de complexidade da imagem ainda é bastante subjetiva e difícil de ser tratada ou prevista digitalmente. Uma definição útil é que a complexidade de uma imagem é uma pontuação da quantidade e da densidade de detalhes ou complexidade das linhas na imagem (PALUMBO et al., 2014).

Este capítulo apresentou e formalizou as diversas manifestações da área da Transferência de Conhecimento no campo da Aprendizagem de Máquina, de acordo com trabalhos que mapeiam o estado da arte nesta linha de pesquisa. Dentre as diferentes configurações

para Transferência de Conhecimento, este trabalho aborda especificamente a Transferência Heterogênea de Aprendizagem, na categoria da Transferência Transdutiva, onde os domínios alvo e fonte diferem mas as tarefas de aprendizagem permanecem compatíveis entre os domínios: $T^s = T^t$ e $D^s \neq D^t$. Mais precisamente esta dissertação trata da Transferência Heterogênea entre os domínios visuais (imagens) e textuais (*tags*). No próximo capítulo será apresentada a arquitetura proposta para realizar uma aprendizagem considerando mutuamente os dois domínios mencionados, isto é, aprender mutuamente entre dados textuais e visuais.

4 ARQUITETURA PROPOSTA

Com o objetivo de explorar a capacidade de transferir conhecimento entre os domínios de textos e imagens, neste capítulo formaliza-se a proposta de uma meta arquitetura de Rede Neural Profunda com a capacidade de ser treinada com dados visuais e dados textuais associados e com a finalidade de sugerir dados textuais para imagens de teste.

Para o treinamento da solução proposta se faz necessário a disponibilidade de um conjunto de imagens e dados textuais associados, *e.g. tags*, descrições ou legendas sobre as imagens, dos quais serão extraídas informações semânticas. Este conjunto multimodal de textos e imagens é considerado como o conjunto de dados do *domínio alvo*, de acordo com as definições de *transferência de conhecimento* já mencionadas. Outra necessidade para o desenvolvimento da arquitetura é a presença de um conjunto expressivo de imagens, no papel de *domínio fonte*, adotado para extrair informações úteis (características) para o *domínio alvo*, onde a quantidade de imagens e textos associados é escassa. Outro conceito fundamental deste trabalho é o de tópico (ou traço) semântico, que de modo superficial pode ser entendido como uma característica específica e compreensível que pode ser observada nas imagens e textos relacionados, explicado na próxima seção.

De modo geral, a arquitetura proposta consiste no treinamento de um conjunto de classificadores individuais binários especializados na predição da ocorrência de cada traço semântico específico presente nas imagens. Os classificadores são treinados com características de alto nível aprendidas das imagens do *domínio fonte* auxiliar, e a composição dos resultados dos classificadores compõem uma predição para os dados de texto que descrevem uma imagem de teste. A Figura 17 apresenta uma visão geral do procedimento de treino da arquitetura proposta, que será tratada em detalhes nas seções deste capítulo.

Os conceitos mais específicos, como o espaço de tópicos semânticos, Classificadores de Tópicos Semânticos e principalmente a definição de tópico semântico, adotados neste trabalho, são explorados em detalhes nas seções subsequentes deste capítulo. A seção final, por sua vez, formaliza toda a arquitetura proposta e apresenta os procedimentos de treino e predição da solução.

4.1 Tópico Semântico

Antes entrar nos detalhes da arquitetura proposta, nesta seção serão explicados os conceitos de tópicos semânticos e espaço de tópicos semânticos adotados neste trabalho.

Para a elaboração do conceito de tópicos semânticos adotado neste trabalho, partiremos do seguinte raciocínio: supondo um conjunto de palavras t_1={avião, planador, aviação, jato, caça, aeronave, nave}, as palavras presentes em t_1 apresentam uma certa afinidade entre si, existe algum significado ou interpretação que une as palavras em t. Os

elementos de t estão de certo modo relacionados com um determinado traço, que pode ser compreendido de modo quase que natural, e este traço pode ser mantido implícito ou pode ser nomeado, seja este nome um dos elementos de t_1 (por exemplo 'aviação') por si só, ou um termo externo específico, por exemplo ('veículos aéreos'). De qualquer modo, um determinado conjunto de palavras pode ser associado a um determinado traço com uma interpretação semântica, e este traço semântico (ou tópico semântico) será um dos objetos de investigação deste estudo.

Tópicos semânticos, neste trabalho, são entendidos como características específicas de alto nível que podem ser derivadas tanto dos dados textuais quanto das características visuais das imagens. Estas características possuem um significado semânticos, sendo assim, podem ser interpretadas pela cognição natural. Por exemplo, supondo 'mar' como um tópico semântico, um segundo conjunto de palavras T_2={mar, oceano, praia, natureza, ilha, onda, surfe} está relacionado com o tópico, bem como imagens de praia, mar, onda, surfe, ilha, oceano e afins estão igualmente relacionadas com o mesmo tópico semântico. As imagens mencionadas tendem a estarem relacionadas entre si pela mesma razão linguística que relaciona as palavras do conjunto t_2 em um mesmo universo semântico. Em termos de visão computacional, as imagens em questão tendem a apresentar características de alto nível similares.

4.1.1 Espaço de Tópicos Semânticos

Bem como definir um espaço latente (características) para as imagens, um espaço textual latente busca tornar mais precisa a representação dos dados textuais para uma definição dos tópicos semânticos.

Um espaço de características latente busca transformar um dado em uma certa distribuição e em certa dimensionalidade para um espaço transformado onde a distribuição dos dados possibilite uma melhor separabilidade dos dados inter-classes, preferencialmente uma separabilidade linear, além de uma redução de dimensionalidade. Uma distribuição com uma melhor separabilidade possibilita uma aprendizagem de máquina mais precisa, principalmente quando a separabilidade for linear, pois funções simples (lineares) já se tornam eficientes para descriminar os dados, em tarefas de classificação por exemplo. Já uma dimensão reduzida possibilita um processamento mais eficiente e menos custoso em termos de memória, sem comprometer gravemente a eficácia do modelo treinado.

Com base nos mesmos conjuntos de exemplo t_1 e t_2, mencionados na seção anterior, definimos que o conjunto t_1 está relacionado com o tópico semântico 'aviação' ao passo que o conjunto t_2 está associado com o tópico semântico 'mar'. Por outro lado, pode-se esperar uma divergência entre os conjuntos t_1 e t_2, dada as distintas naturezas de seus elementos. A definição de um espaço textual latente, ou seja, um conjunto comum entre os conjuntos de palavras originais busca tornar possível uma quantização precisa

da proximidade entre os conjuntos, e assim, possibilitar uma aprendizagem eficiente dos tópicos semânticos associados aos dados textuais.

4.2 Classificadores de Tópicos Semânticos

Como descrito na seção anterior, um espaço de características latente busca definir um espaço onde dados sejam representados de forma mais eficiente para uma determinada finalidade. No domínio de imagens, características latentes podem ser aprendidas, por exemplo, para uma atividade de classificação.

Figura 15 – Quadro com Exemplo de Imagem Completa

Fonte: (LIN et al., 2014)

Neste trabalho, o raciocínio que embasa a definição dos Classificadores de tópicos semânticos é justamente a aprendizagem de modelos especializados em problemas binários de classificação de imagens. Um espaço de características visuais latente é aprendido para as imagens, enquanto que os problemas de classificação são sobre a ocorrência de cada tópico semântico possível. Ou seja, para determinar cada tópico semântico presente em uma imagem, um modelo de classificação de imagens é aprendido a partir do espaço de características visuais.

Na presente proposta, os tópicos semânticos devem ser aprendidos individualmente, sendo assim, um modelo deve ser treinado para cada tópico semântico de interesse. Isto é, assumindo λ tópicos semânticos, teremos λ Classificadores de Tópicos Semânticos. Uma primeira justificativa para essa abordagem é a redução de um problema mais complexo

em λ problemas binários, provavelmente mais simples. Supondo a imagem da Figura 15, podemos observar diversos elementos distintos na mesma imagem, consequentemente, diferentes possíveis tópicos semânticos, a exemplificar: “gato”, “pássaro”, “rio”, “barco”, “pessoas”, “natureza”, “edificações” e “árvore”.

Existem regiões específicas na Figura 15 que justificam a existência de cada tópico semântico dado como exemplo, como pode ser observado na Figura 16. Propor um modelo único para identificar cada um dos traços semânticos, incluindo aqueles não presentes na imagem (por exemplo “carro” e “aviação”), significa treinar um modelo em um espaço de busca extremamente complexo. Por outro lado, treinar um modelo particular para decidir sobre a ocorrência ou não (problema binário) de cada um dos tópicos semânticos representa um treinamento em um espaço de busca menos complexo.

Outro ponto positivo desta abordagem é a capacidade de paralelismo do treinamento, uma vez que cada Classificador de Tópicos Semânticos pode ser aprendido individualmente, otimizando consideravelmente o tempo de treino da arquitetura proposta.

Figura 16 – Quadro com Exemplos de Tópicos Semânticos em Imagem - Recortes

Fonte: Wanderley (2018)

4.3 Modelo Generativo para Dados Textuais

Para o treinamento da arquitetura, um espaço de tópicos semânticos $\mathbb{S}$ deve ser definido a partir do espaço textual $\mathbb{T}$. O objetivo do modelo generativo Λ proposto é de realizar dois papeis fundamentais na técnica proposta: primeiramente o modelo generativo Λ auxilia a identificação do vetor de tópicos semânticos $\mathbf{s}$ a partir dos dados textuais originais $\mathbf{t}$. O segundo papel do modelo Λ é utilizar a capacidade generativa deste ($\Lambda^{-1}(.)$) para sugerir dados textuais $\hat{\mathbf{t}}$ a partir de um vetor de tópicos semânticos $\mathbf{s}$.

Seguindo a mesma intuição do conjunto de classificadores específicos para identificar cada um dos tópicos semânticos nas imagens, o modelo generativo Λ baseia-se em um conjunto de modelos generativos igualmente específicos para identificação dos tópicos semânticos nos dados textuais. Desse modo, o modelo Λ deve ser treinado com os dados textuais de interesse e com os respectivos tópicos semânticos, a predição do modelo Λ resulta no vetor de tópicos semânticos $\mathbf{s}$ identificados para um dado textual $\mathbf{t}$, conforme a Equação 4.1. Já a reconstrução de um vetor de dados textuais $\hat{\mathbf{t}}$ a partir dos tópicos semânticos $\mathbf{s}$ se dá conforme a Equação 4.2.

$$\mathbf{s} = \Lambda(\mathbf{t}) \tag{4.1}$$

A regra específica para definir quais são os tópicos semânticos $\mathbb{S}$ presentes no vocabulário do espaço textual original $\mathbb{T}$ fica entendida como uma particularidade de implementação da técnica proposta neste capítulo. No próximo capítulo será proposta uma regra possível para esta finalidade, mas outras regras também podem ser exploradas.

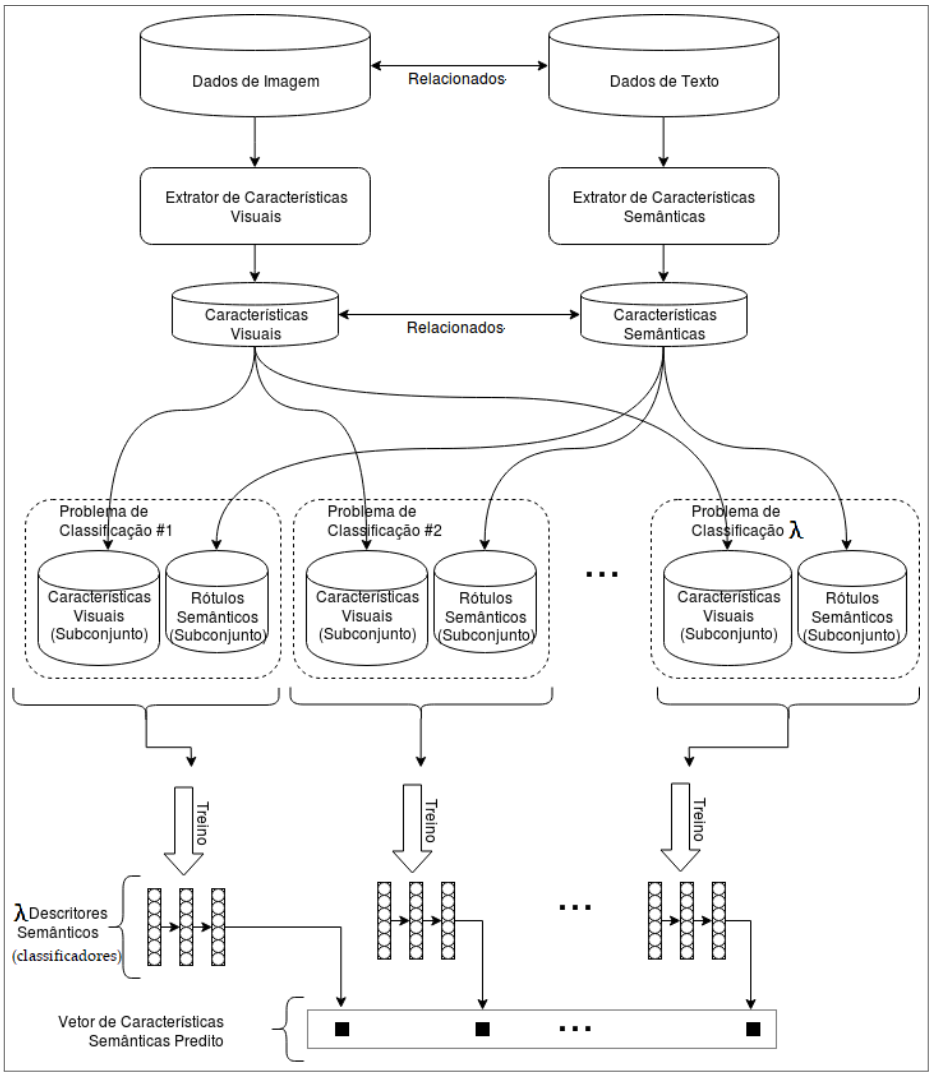
4.4 Arquitetura Proposta: Visão Geral e Algoritmos

O modelo proposto consiste em uma meta arquitetura de Rede Neural Profunda com a capacidade de ser treinada com pares de imagens e textos associados. Como saída, a arquitetura proposta retorna uma sugestão para os dados textuais associados às imagens de entrada. Internamente, a técnica utiliza-se de extratores de características, tanto visuais quanto textuais, para auxiliar na aprendizagem dos tópicos semânticos presentes nas imagens. Cada tópico semântico é aprendido individualmente, resultando em classificadores individuais especializados na identificação de cada tópico semântico existente em uma imagem. A Figura 17 apresenta uma visão geral do procedimento de treino da arquitetura proposta.

A quantidade de tópicos semânticos aprendidos pelo modelo, e consequentemente a quantidade de classificadores internos ao modelo, é definida pelo hiper-parâmetro λ. Assim, o tamanho do espaço semântico $\mathbb{S}$ fica definido pela quantidade λ de tópicos semânticos escolhidos, ou seja, dado um vetor de tópicos semânticos $\mathbf{s}$, temos que $\mathbf{s} = (s^1, s^2, ...s^l, ..., s^\lambda), \mathbf{s} \in \mathbb{S}$. Os λ tópicos semânticos a serem aprendidos podem ser derivados do próprio espaço textual $\mathbb{T}$ ou alguma regra alternativa, o fundamental é que os tópicos semânticos sejam traços presentes em ambos os domínios, e não uma particularidade de um dos domínios – textual ou visual.

O Algoritmo 1 apresenta o procedimento de treino do modelo. A entrada do algoritmo é conjunto de pares de imagens e textos associados $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^n$, com n instâncias disponíveis, onde $\mathbf{x} \in \mathcal{X}$, sendo $\mathcal{X}$ o espaço de características visuais e $\mathbf{t} \in \mathbb{T}$, enquanto que a saída do treinamento é o modelo composto por um conjunto de λ classificadores de tópicos semânticos $\mathbb{C}$. Para cada tópico semântico $\{s^l\}_{l=1}^\lambda$ no espaço semântico $\mathbb{S}$, são

Figura 17 – Diagrama de Treino da Arquitetura Proposta



Fonte: Wanderley (2018)

selecionadas as imagens as quais os dados textuais apresentam o tópico semântico de interesse s^l para compor a partição positiva (rótulos positivos $\mathbf{D}_\mathbf{l}^+$) do conjunto de imagens de treino $\mathbf{D_l}$ para o l-ésimo tópico semântico. Para compor a partição negativa $\mathbf{D}_\mathbf{l}^-$ (de forma balanceada) de $\mathbf{D_l}$, são selecionadas imagens aleatórias de $\mathcal{D}$ as quais os dados textuais associados não incluem o tópico semântico s^l. Uma vez que o conjunto de imagens de treino está definido para o l-ésimo tópico semântico, um classificador binário (Classificador de Tópicos Semânticos), C_l é treinado e incluído ao conjunto de classificadores $\mathbb{C}$ a ser retornado.

Algoritmo 1: Algoritmo de Treino

Entrada:

$\mathbf{D} = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^{n}$: Conjunto de imagens e textos associados

Resultado:

$\mathbb{C}$: Conjunto de classificadores binários de tópicos semânticos

inicialização:

$\mathbb{C} \leftarrow \{\}$

para cada tópico semântico s^l **faça:**

  $\mathbf{D}_l^+ \leftarrow \{(\mathbf{x}_i, s_i^l) | s_i^l = 1\}_{i=1}^{n}$ *//Casos positivos, i.e., imagens para as quais o texto associado contém o l-ésimo tópico semântico* s^l.

  $\mathbf{D}_l^- \leftarrow \{(\mathbf{x}_i, s_i^l) | s_i^l = 0\}_{i=1}^{n}$ *//Casos negativos, i.e., imagens para as quais o texto associado não contém o l-ésimo tópico semântico, limitado ao mesmo tamanho de* $\mathbf{D}_l^+$.

  $\mathbf{D}_l \leftarrow \mathbf{D}_l^+ \cup \mathbf{D}_l^-$

  Treine o classificador C_l usando $\mathbf{D}_l$

  $\mathbb{C} \leftarrow \mathbb{C} \cup \{C_l\}$

fim para

retorne $\mathbb{C}$

No Algoritmo 2 é descrito o procedimento de predição do modelo. Como entrada temos uma instância de imagem de teste $\mathbf{x}$, e como retorno do algoritmo temos o vetor de dados textuais $\hat{\mathbf{t}}$ sugeridos para $\mathbf{x}$. Internamente o modelo sugere os tópicos semânticos $\hat{\mathbf{s}}$ sugeridos para $\mathbf{x}$. O procedimento para formar $\hat{\mathbf{s}}$ consiste em estimar cada tópico semântico s^l, parte de $\hat{\mathbf{s}}$ com base na predição de cada classificador C_l do conjunto de classificadores $\mathbb{C}$, para cada tópico semântico $s^l, s^l \in (s^1, ..., s^l, ..., s^\lambda)$.

Algoritmo 2: Algoritmo de Predição

Entrada:

$\mathbf{x}$: Imagem instância de teste

$\mathbb{C}$: Conjunto de classificadores binários de tópicos semânticos

Λ: Modelo generativo tradutor de espaço textual para espaço semântico

Resultado:

$\hat{\mathbf{t}}$: Vetor de dados textuais sugeridos para uma instância de teste

inicialização:

$\hat{\mathbf{s}} \leftarrow$ vetor nulo no espaço $\mathbb{S}$

para cada classificador $\{C_l\}_{l=1}^{\lambda}$ em $\mathbb{C}$ **faça:**

  $\hat{s}^l \leftarrow C_l(\mathbf{x})$

fim para

$\hat{\mathbf{t}} \leftarrow \Lambda^{-1}(\hat{\mathbf{s}})$

retorne $\hat{\mathbf{t}}$

Desta forma, o modelo treinado apresenta a capacidade de estimar um vetor de características semânticas $\hat{\mathbf{s}}_i$ para uma imagem de teste $\mathbf{x}_i$. Por fim, com o objetivo de manter a mesma distribuição do espaço textual $\mathbb{T}$, um modelo generativo $\Lambda(.)$ é adotado para aprender a distribuição de $\mathbb{T}$ com base na distribuição de tópicos semânticos do espaço semântico $\mathbb{S}$, e assim, recuperar um vetor de dados textuais estimado $\hat{\mathbf{t}}$ a partir do vetor estimado $\hat{\mathbf{s}}$ de tópicos semânticos, aplicando-se $\Lambda^{-1}(.)$, conforme a Equação 4.2.

$$\hat{\mathbf{t}} = \Lambda^{-1}(\hat{\mathbf{s}}) \tag{4.2}$$

4.5 Protótipo Implementado

A arquitetura proposta trata-se de uma meta-arquitetura, e portanto pode ser implementada de diferentes formas, a depender dos componentes usados para implementação. Nesta seção é apresentada uma implementação particular para a arquitetura proposta, a qual será avaliada em experimentos esmiuçados no próximo capítulo. Esta seção concentra-se em detalhar uma implementação possível para a arquitetura proposta, mas não limitar o entendimento da proposta à uma configuração específica. Ao longo das próximas seções, alternativas de implementação para cada componente são mencionadas.

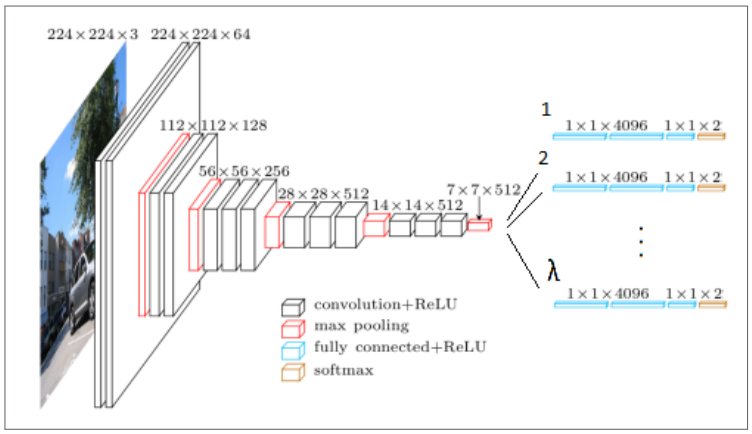
De modo geral, o protótipo implementado envolve o uso de um extrator de características visuais baseado nas camadas convolucionais de uma VGG-16 (SIMONYAN; ZISSERMAN, 2014) pré-treinada na base de imagens *ImageNet* (RUSSAKOVSKY et al., 2015), além de um extrator de características textuais baseada na combinação de classificadores *Naive Bayes* (MCCALLUM; NIGAM et al., 1998). A definição do espaço semântico $\mathbb{S}$, bem como a escolha dos λ tópicos semânticos, é feita com base em uma regra específica, nomeada "Regra dos Termos Mais Frequentes", também explicada nesta seção. E por fim, uma combinação de redes neurais no papel dos Descritores de Características Semânticas.

O protótipo implementado foi treinado para tarefas de reconhecimento na base de dados NUS-WIDE (CHUA et al., 2009). Em termos gerais, a base de dados NUS-WIDE é uma base aberta de imagens e textos associados coletada do site de hospedagem e compartilhamento de fotografias, desenhos e ilustrações *Flickr* (FLICKR; LUDICORP; YAHOO!, 2004). As imagens presentes no *Flickr* apresentam (em sua maioria) um conjunto de dados textuais no formato de rótulos – *tags*, e são justamente alguns desses pares de imagens e conjunto de *tags* associadas que compõem a base de dados NUS-WIDE. O Capítulo 5 apresenta um estudo mais detalhado sobre a base de dados NUS-WIDE.

4.5.1 Conjunto de Classificadores de Tópicos Semânticos

Esta seção descreve a implementação do conjunto de classificadores $\mathbb{C}$ usado para predição dos tópicos semânticos de uma imagem de teste, de acordo com o Algoritmo 1, que define que para cada tópico semântico s_l em $\mathbb{S}$ um classificador binário específico C_l seja treinado.

Figura 18 – Arquitetura do modelo implementado inspirado na arquitetura original da VGG-16



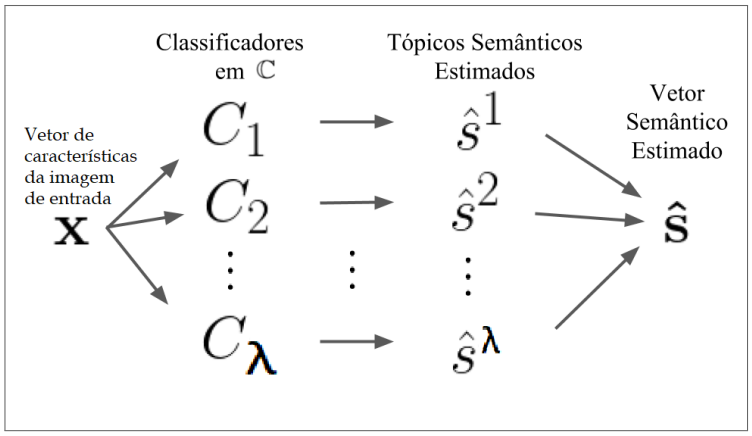
Fonte: Adaptação de Blier (2016)

Para o protótipo avaliado, o classificador implementado é uma rede neural baseada nas três últimas camadas totalmente conectadas da VGG-16, uma vez que a pate convolucional da VGG-16 foi adotada como codificador. O protótipo apresenta igualmente três camadas totalmente conectadas, diferindo pela quantidade de neurônios utilizados. Conforme a arquitetura original da VGG-16, as três últimas camadas são totalmente conectadas (ou também chamadas de camadas densas), contendo respectivamente 4096, 4096 e 1000 neurônios, sendo as primeiras duas camadas com ativação ReLU (GLOROT; BORDES; BENGIO, 2011) e a última camada utilizando a ativação *SoftMax*, para a finalidade de classificação (SIMONYAN; ZISSERMAN, 2014). Sendo assim, buscando aproximar-se da arquitetura original da VGG-16, os classificadores adotados consistem em redes contendo três camadas, além da camada de entrada no formato de 7x7x512. As duas primeiras camadas na configuração dos classificadores seguem conforme a arquitetura original, 4096 neurônios com ativação ReLU, apenas a última camada foi modificada para 2 neurônios com ativação *SoftMax*, uma vez que a tarefa de cada classificador particular é uma classificação binária. Finalmente, λ classificadores binários são treinados, um para cada tópico semântico s_l em $\mathbb{S}$, conforme o Algoritmo 1, a Figura 18 ilustra a arquitetura do modelo implementado, inspirada na arquitetura original da VGG-16.

Conforme descrito no Algoritmo 2, a predição do vetor de tópicos semânticos $\hat{\mathbf{s}}$ a partir das características de alto nível de uma imagem $\mathbf{x}$ de teste é feita combinando a predição de cada tópico semântico s_l por cada classificador C_l particular no conjunto de classificadores $\mathbb{C}$. A Figura 19 ilustra o fluxo de dados para a composição de $\hat{\mathbf{s}}$ a partir da aplicação do modelo em $\mathbf{x}$.

Vale salientar que o método proposto, por ser baseado na combinação de diferentes classificadores especializados, apresenta uma maior capacidade em aprender e identificar características visuais específicas para diferentes traços semânticos simultaneamente. Como pode ser observado na Figura 16, diferentes elementos podem ser identificados por

Figura 19 – Fluxo de dados durante a predição do modelo



Fonte: Wanderley (2018)

cada um dos classificadores treinados no modelo completo, como por exemplo, a presença de gatos ou cachorros indiciam a ocorrência dos tópicos semânticos homônimos, bem como a presença de construções ou elementos naturais evidenciam a presença de outros tópicos semânticos relacionados. Sendo assim, um modelo da arquitetura proposta não se limita a classificar unicamente uma imagem, mesmo quando diferentes elementos semânticos estejam presentes na mesma.

Finalmente, o classificador adotado para compor o conjunto de classificadores no protótipo implementado também pode ser variado. Diversas são as alternativas de classificadores que podem ser adotadas para esta finalidade, contudo, esta linha de investigação reserva-se a possíveis trabalhos futuros.

4.5.2 Extrator de Características Visuais

Conforme explicado na descrição da técnica proposta, objetivo de utilizar um extrator de características visuais consiste em possibilitar uma aprendizagem mais eficiente e precisa ao desenvolver o treinamento em um espaço de características reduzido e com maior capacidade representativa sobre os dados visuais *i.e.* imagens. Esta seção descreve qual extrator de características visuais foi adotado para o protótipo implementado, como funciona e como esse extrator se aplica ao protótipo e quais são as características de alto nível retornados pelo extrator. De acordo com a categoria de Redes Neurais Convolucionais de Auto-Codificadores, o extrator de características também pode ser referido como um codificador, pois trata-se da mesma arquitetura de Rede Neural Profunda.

O codificador adotado para a implementação do protótipo foi a parte convolucional de uma VGG-16 pré-treinada na base de imagens *ImageNet* (RUSSAKOVSKY et al., 2015). A VGG é uma arquitetura de rede neural convolucional proposta por Simonyan e Zisserman (2014) no trabalho "*Very Deep Convolutional Networks for Large-Scale Image Recognition*" , solução ganhadora do primeiro e segundo lugar da Competição de Reconhecimento

Visual de Larga Escala ImageNet – ILSVRC – na edição do ano de 2014 (RUSSAKOVSKY et al., 2015).

A VGG-16 é a versão da VGG com 16 camadas, sendo as primeiras 13 camadas convolucionais e as 3 últimas camadas densas – totalmente conectadas (SIMONYAN; ZISSERMAN, 2014). A função de ativação nas camadas da rede VGG é a função de ativação ReLU (GLOROT; BORDES; BENGIO, 2011; LECUN et al., 2012), exceto a última camada, que utiliza uma ativação Softmax (BISHOP, 2006). A configuração detalhada das camadas da VGG-16 pode ser vista no Capitulo 2, a Figura 8 ilustra a arquitetura geral da VGG-16.

As camadas convolucionais da VGG-16, *i.e.*, as primeiras 13 camadas, foram adotadas como o extrator de características visuais (também chamado de codificador) para o protótipo implementado. A saída do codificador convolucional da VGG-16 é uma representação em alto nível e de menor dimensionalidade das imagens de entrada. Para a entrada originalmente usada na VGG-16, no formato de $224 \times 224 \times 3$ (224 *pixels* de altura por 224 *pixels* de largura em 3 canais de cores – *Red* (vermelho), *Green* (verde) e *Blue* (azul) (RGB)), a saída da última camada do codificador convolucional é uma representação da imegem de entrada no formato $7 \times 7 \times 512$ (7 *pixels* de altura por 7 *pixels* de largura em 512 canais). A redução do codificador representa a transformação de uma entrada de 150528 dimensões ($224 \times 224 \times 3$) para uma representação em 25088 dimensões ($7 \times 7 \times 512$), ou seja, uma redução de mais de 80% da dimensionalidade original. Os detalhes das operações de pré-processamento realizadas nos dados utilizados nos experimentos são descritos no Capítulo 5, bem como as estratégias de refinamento de modelo aplicadas.

A VGG-16 adotada reusa os pesos aprendidos após o treinamento na base de imagens *ImageNet* e com os pesos das ultimas camadas totalmente conectadas refinados na base de dados NUS-WIDE, de acordo com as definições de transferência de aprendizagem mencionadas no Capítulo 3. Dado o volume de imagens presentes na base *ImageNet*, o domínio fonte auxiliar deste trabalho, espera-se que os pesos pré-treinados dos codificadores apresentem um bom desempenho nas atividades de reconhecimento e poupe tempo de treinamento nos problemas de reconhecimento dos tópicos semânticos específicos (YOSINSKI et al., 2014).

Por fim, é importante lembrar que a adoção do codificador convolucional da VGG-16 pré-treinado na base de imagens *ImageNet* é apenas uma possibilidade de implementação para a arquitetura proposta, escolhida por ser uma das arquiteturas e modelos pré-treinados mais utilizado atualmente, além de apresentar resultados satisfatórios em diversas aplicações de imagens. No entanto, outras arquiteturas podem ser aplicadas para a mesma finalidade, como por exemplo a VGG-19 (SIMONYAN; ZISSERMAN, 2014), *ResNet* (HE et al., 2016), *MobileNet* (HOWARD et al., 2017), *Inception* (SZEGEDY et al., 2015) entre outras. Inclusive os pesos podem ser aprendidos do treinamento em outras bases de imagens. Investigar essas variações fica entendido como atividades de um possível trabalho futuro.

4.5.3 Regra dos Termos Mais Frequentes – *Top Termo*

Nas seções iniciais foi exposta a necessidade de se definir os tópicos semânticos a partir do vocabulário dos dados textuais originais. Nesta seção é apresentada uma regra pra desempenhar essa atividade baseada nos termos mais frequentes dentre os dados textuais, denominada Regra dos Termos Mais Frequentes. A regra proposta tem como objetivo definir um vocabulário de tópicos semânticos $\mathbb{S}$ a partir do vocabulário original $\mathbb{T}$, e por consequência, definir também um vocabulário reduzido $\mathbb{T}_s$. A união dos vocabulários $\mathbb{T}_s$ e $\mathbb{S}$ resultam no vocabulário original dos dados textuais $\mathbb{T}$, conforme a Equação 4.3.

Conforme explicado na seção sobre Tópicos Semânticos, para um conjunto de palavras, termos ou *tags* que descrevam imagens, subconjuntos desses termos podem estar relacionados entre si por motivos semânticos. No caso da base de dados objeto de estudo deste trabalho, a base NUS-WIDE (CHUA et al., 2009), cada imagem apresenta um conjunto de termos descritores associados. A distribuição dos termos descritores indiciam que alguns termos possuem maior frequência que outros, principalmente os termos que apontam características mais genéricas, como "gato" ou "carro". A regra detalhada nesta subseção baseia-se na identificação dos termos mais frequentes em um dado vocabulário (conjunto de termos) $\mathbb{T}$ para a definição de um espaço de termos semânticos $\mathbb{S}$. Denominada como Regra dos Termos Mais Frequentes, onde os λ termos de maior frequência são denominados como *Top Termos*, a regra assume que os termos mais frequentes são aqueles que abrangem os principais elementos possíveis presentes nas descrições das imagens e assim compõem o espaço semântico $\mathbb{S}$. Por consequência, os termos restantes aos *Top Termos* $\mathbb{S}$ constitui um vocabulário reduzido e mais específico $\mathbb{T}_s$ que aponta para os mesmos traços semânticos presentes em $\mathbb{S}$. Diante disso, podemos formalizar que o vocabulário original $\mathbb{T}$ pode ser composto pela união dos *Top Termos* e os termos restantes $\mathbb{T}_s$, ou seja:

$$\mathbb{T} = \mathbb{S} \cup \mathbb{T}_s \tag{4.3}$$

O procedimento da Regra dos Termos Mais Frequentes para definir $\mathbb{S}$ e $\mathbb{T}_s$ a partir do vocabulário original $\mathbb{T}$ consiste basicamente em ordenar os temos em $\mathbb{T}$ por ordem decrescente de ocorrência e então selecionar os λ termos mais frequentes para compor $\mathbb{S}$. O restante dos termos (os termos menos frequentes) compõem $\mathbb{T}_s$. O Algoritmo 3 formaliza a regra proposta.

Algoritmo 3: Algoritmo da Regra dos Termos Mais Frequentes

Entrada:

$\mathbb{T}$: Vocabulário do espaço textual original

$\mathbf{T}$: Dados textuais

λ: Quantidade de *Top Termos* de interesse

Resultado:

$\mathbb{S}$: Vocabulário do espaço semântico

$\mathbb{T}_s$: Vocabulário do espaço textual reduzido

1 $\tau \leftarrow$ contagem da ocorrência dos termos em $\mathbb{T}$ de acordo com a distribuição de $\mathbf{T}$

2 $\tau \leftarrow$ ordenação decrescente de τ de acordo com a ocorrência dos termos

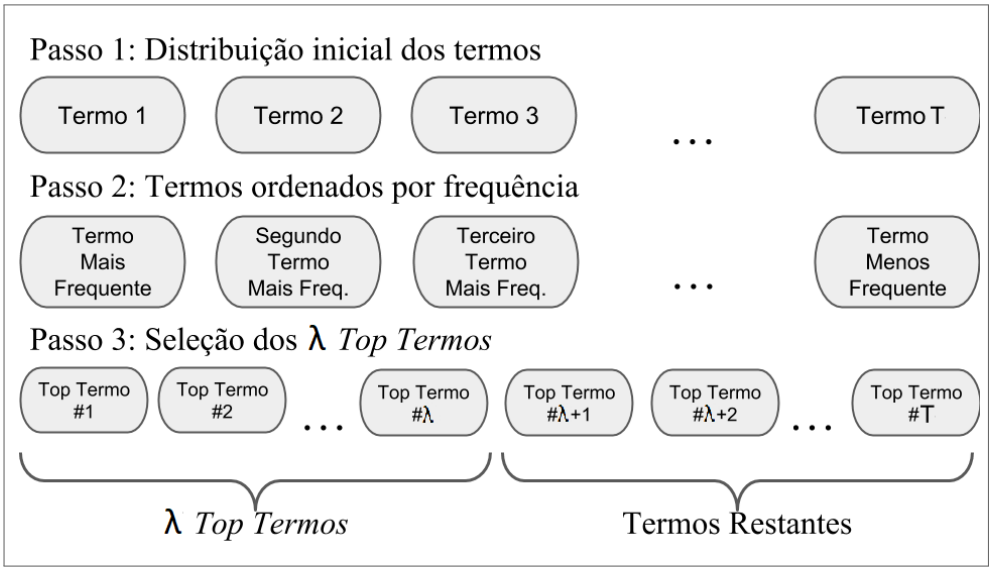
3 $\mathbb{S} \leftarrow \lambda$ termos mais frequentes, *i.e.*, primeiro termo em τ até o λ-ésimo termo em τ

4 $\mathbb{T}_s \leftarrow$ termos menos frequentes restantes, *i.e.*, do $(\lambda + 1)$-ésimo termo em τ até o último termo em τ

5 **retorne** $\mathbb{S}$ e $\mathbb{T}_s$

A Figura 20 ilustra os passos para compor o espaço semântico $\mathbb{S}$ (para λ *Top Termos*) e o espaço textual reduzido restante $\mathbb{T}_s$ aplicando-se a Regra dos Termos Mais Frequentes no vocabulário original $\mathbb{T}$.

Figura 20 – Ilustração dos passos da Regra dos Termos Mais Frequentes



Fonte: Wanderley (2018)

Exemplificando o raciocínio apresentado, vamos supor um vocabulário com 1000 termos provenientes da descrição de filmes. Ao definir os λ Top Termos, sendo $\lambda = 5$, teríamos $\mathbb{S}_{\text{exemplo}} = \{$"drama", "comédia", "romance", "guerra", "ação"$\}$, pois estes seriam os 5 termos mais presentes nas descrições dos filmes. Já os termos {"Segunda Guerra", "Dia-D", "Normandia", "1944", "batalha"} seriam parte do espaço textual restante e mais

específico $\mathbb{T}_{s\text{exemplo}}$, o qual provavelmente remete ao *Top Termo* "guerra", porém trata de um conceito mais específico dentre as descrições relacionadas ao mesmo *Top Termo*.

Por exemplo, para reconstruir os termos presentes na descrição original, de acordo com a Equação 4.3, teríamos $\mathbb{T}_{\text{exemplo}} = \mathbb{S}_{\text{exemplo}} \cup \mathbb{T}_{s\text{exemplo}} =$ {"guerra", "Segunda Guerra", "Dia-D", "Normandia", "1944", "batalha"}.

4.5.4 Extrator de Características Textuais

Com a mesma motivação descrita na seção anterior para a definição de um espaço visual latente para o domínio de imagens, nesta seção será detalhada como o domínio textual $\mathbb{T}$ é derivado em dois domínios reduzidos, são eles o domínio de tópicos semânticos $\mathbb{S}$ e o domínio textual restante $\mathbb{T}_s$. Também será detalhada a implementação do modelo generativo Λ, que busca aprender a distribuição de $\mathbb{T}_s$ e auxiliar na reconstrução das predições no domínio textual original $\mathbb{T}$ a partir da predição realizada no espaço semântico $\mathbb{S}$. Nesta seção será detalhado como derivamos o espaço textual original $\mathbb{T}$ no espaço de tópicos semântico $\mathbb{S}$ e no espaço textual reduzido $\mathbb{T}_s$.

Na base de dados NUS-WIDE, os dados textuais associados às imagens disponíveis apresentam-se de duas formas. A primeira é um conjunto com as palavras originais (*tags*) presentes na descrição de cada imagem. Enquanto que a segunda forma consiste em um vetor binário de 1000 posições (*bag-of-words*) o qual representa quais dentre as 1000 palavras mais comuns usadas nas descrições das imagens estão presentes na descrição de cada imagem em particular (CHUA et al., 2009). Para o protótipo implementado, será adotada a segunda representação do domínio textual, representado como um vetor de 1000 posições (1000-dimensional) onde cada posição representa a ocorrência ou ou não de cada uma das 1000 palavras mais frequentes nas descrições das imagens.

O espaço de tópicos semânticos $\mathbb{S}$ é definido de acordo com a Regra dos Termos Mais Frequentes, previamente apresentada. Esta regra determina λ termos mais frequentes dentre o espaço textual $\mathbb{T}$, sendo $\mathbb{T} = \{t^1, t^2, ...t^T\}$, onde $\mathbb{T}$ possui T dimensões. Para este estudo, temos $T = 1000$, em virtude dos dados textuais T-dimensionais da base NUS-WIDE, conforme mencionado. Deste modo, o modelo generativo Λ é usado para transformar instâncias $\mathbf{t}$ no espaço textual $\mathbb{T}$, para uma representação $\mathbf{s}$ no espaço textual semântico $\mathbb{S}$, de acordo com a Equação 4.1, sendo $\mathbb{S} = \{s^1, s^2, ..., s^\lambda\}$, para λ tópicos semânticos.

Excluindo-se os termos mais frequentes $\mathbb{S}$ do espaço textual original $\mathbb{T}$, temos um espaço textual reduzido resultante $\mathbb{T}_s$, como será visto em detalhes na descrição da Regra dos Termos Mais Frequentes. Sendo assim, os dados textuais $\mathbf{t}$ no espaço textual original $\mathbb{T}$, $\mathbf{t} \in \mathbb{T}$, é transformado para uma representação reduzida $\mathbf{t}_s$ no espaço resultante $\mathbb{T}_s$, $\mathbf{t}_s \in \mathbb{T}_s$. Finalmente, aplicando-se o modelo generativo Λ no espaço textual reduzido $\mathbb{T}_s$,

a Equação 4.1 se torna:

$$\mathbf{s} = \Lambda(\mathbf{t}_s) \tag{4.4}$$

O modelo generativo Λ implementado para o presente protótipo utilizou também um raciocínio de combinação de modelos. Sendo $\mathbb{G}$ um conjunto de λ modelos generativos G, um para cada tópico semântico s^l em $\mathbb{S}$, *i.e.* $\mathbb{G} = \{G^1, G^2, ..., G^l, ..., G^\lambda\}$, o modelo generativo Λ foi implementado com base na combinação de modelos $\mathbb{G}$. Cada modelo generativo G_λ consiste em um modelo Bernoulli Naive Bayes (MCCALLUM; NIGAM et al., 1998), treinados no espaço textual reduzido $\mathbb{T}_s$ com o objetivo de predizer cada um dos tópicos semânticos s^l particular em $\mathbf{s}$, seguindo algoritmos equivalentes aos Algoritmos 1 e 2.

A capacidade generativa do modelo Λ é usada para a reconstrução dos dados textuais $\hat{\mathbf{t}}$, a partir do vetor de características semânticas $\hat{\mathbf{s}}$ estimado pelo Algoritmo 2. Sendo assim, define-se a aproximação dos dados textuais $\hat{\mathbf{t}}$ para um vetor semântico estimado $\hat{\mathbf{s}}$ de acordo com a seguinte equação:

$$\hat{\mathbf{t}} = \hat{\mathbf{s}} \cup \hat{\mathbf{t}_s} = \hat{\mathbf{s}} \cup \Lambda^{-1}(\hat{\mathbf{s}}) \tag{4.5}$$

Conforme descrito, para o modelo generativo do protótipo implementado foi adotada uma combinação de modelos Bernoulli Naive Bayes, mas outras abordagens generativas podem ser investigadas para esta finalidade, *e.g.*, *Restricted Boltzmann Machine* (RBM) (SUTSKEVER; HINTON; TAYLOR, 2009) e DBN (HINTON, 2009). Ainda como alternativa de implementação, a definição do espaço semântico $\mathbb{S}$ realizada baseou-se na Regra dos Termos Mais Frequentes, proposta a seguir, mas pode-se investigar regras alternativas, por exemplo alguma regra baseada em ontologias. Mas este tema também fica entendido como evoluções de um trabalho futuro.

5 EXPERIMENTOS E RESULTADOS

Neste capítulo são detalhados os experimentos realizados para avaliar o desempenho da arquitetura proposta. A primeira seção apresenta um estudo exploratório preliminar sobre a base de imagens e textos associados utilizada para a realização dos experimentos, a base de dados NUS-WIDE (CHUA et al., 2009). As seções seguintes apresentam em detalhes os experimentos realizados, descrevendo os objetivos, método experimental, resultados obtidos além de uma discussão sobre os resultados obtidos. Em linhas gerais, os experimentos realizados buscam avaliar o desempenho da arquitetura proposta em três diferentes aspectos:

1. **Identificação dos tópicos semânticos:** este experimento avalia a capacidade de aprendizagem dos tópicos semânticos em $\mathbb{S}$ pelos classificadores membros do conjunto de classificadores $\mathbb{C}$, interno ao modelo. Cada classificador em $\mathbb{C}$ é um classificador binário com o objetivo de detectar a existência de um traço semântico específico, sendo assim, este experimento considera a acurácia como medida de desempenho dos classificadores na atividade de classificar imagens de teste quanto à presença de tópicos semânticos específicos.

2. **Reconstrução de dados textuais:** este experimento avalia a capacidade generativa do modelo em estimar, para uma imagem de teste, os dados textuais $\hat{\mathbf{t}}$ no vocabulário $\mathbb{T}$ inicial com relação aos dados textuais originais $\mathbf{t}$. Como medidas de desempenho neste experimento, foram adotadas a medida de Precisão, Revocação (*Precision* e *Recall*) e a similaridade de Jaccard.

3. **Classificação de imagens:** este experimento avalia a eficiência dos dados textuais sugeridos pelo modelo proposto quando utilizados para uma atividade de classificação de imagens. A finalidade da arquitetura proposta não é necessariamente a classificação de imagens, no entanto, é pertinente avaliar a qualidade da saída do modelo proposto como codificador para uma tarefa de classificação de imagens. A acurácia também é adotada como medida de desempenho, inclusive para comparar os resultados do modelo proposto com outras abordagens similares, presentes na literatura.

5.1 Análise Descritiva da Base de Dados NUS-WIDE

Esta seção apresenta um estudo exploratório sobre a base de dados NUS-WIDE. Trata-se de uma base aberta de imagens e textos associados coletada do site de hospedagem e compartilhamento de fotografias, desenhos e ilustrações *Flickr* (FLICKR; LUDICORP;

Tabela 1 – Descrição da base de dados NUS-WIDE

	Quantidade de *Tags* por Descrição			
Quantidade de Instâncias	Máx.	Mín.	Méd.	Desv. Padr.
269648	131	0	5,78	4,82
Tamanho do Vocabulário	**Ocorrências das *Tags* nas Descrições**			
(quantidade de *Tags*)	Máx.	Mín.	Méd.	Desv. Padr.
1000	20142	545	1559,46	1996,30

Tabela 2 – Descrição da partição com 10 classes da base de dados NUS-WIDE

	Quantidade de *Tags* por Descrição			
Quantidade de Instâncias	Máx.	Mín.	Méd.	Desv. Padr.
3253	41	1	6,45	4,41
Tamanho do Vocabulário	**Ocorrências das *Tags* nas Descrições**			
(quantidade de *Tags*)	Máx.	Mín.	Méd.	Desv. Padr.
1000	493	0	20,98	49,18

YAHOO!, 2004). As imagens presentes no *Flickr* apresentam (em sua maioria) um conjunto de dados textuais no formato de rótulos – *tags*, e são justamente alguns desses pares de imagens e conjunto de *tags* associadas que compõem a base de dados NUS-WIDE (CHUA et al., 2009).

As imagens da base NUS-WIDE são imagens coloridas, ou seja, apresentam três canais de cores RGB. Além disto, as imagens apresentam larguras e alturas variadas. Para os experimentos, em virtude da entrada fixa do modelo, as imagens da base foram submetidas a uma operação de redimensionamento para o formato fixo de 224 *pixels* de altura, 224 *pixels* de largura e 3 canais de cores – $224 \times 224 \times 3$. Os dados textuais da base usados neste trabalho são os mil termos mais frequentes, segundo os autores da base. Sendo assim, a dimensionalidade (tamanho do vocabulário) dos dados textuais é igual a 1000.

A Tabela 1 apresenta uma descrição dos dados da base NUS-WIDE. A base completa é constituída por 269648 instâncias, pares de imagens e textos descritivos, onde em média cada descrição apresenta 5,78 termos (ou *Tags*). Em média, cada descrição apresenta 1559,46 *Tags*, mas um elevado desvio padrão foi observado: 1996,30. A Figura 21 apresenta a distribuição da quantidade de *Tags* por descrição (21a) e a distribuição das ocorrências das *Tags* nas descrições (21b).

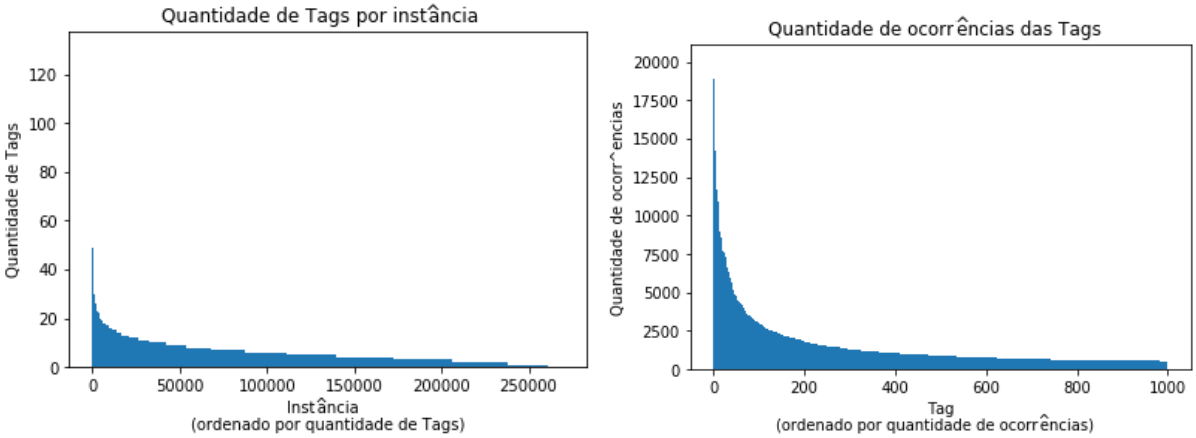
Para os experimentos foi utilizada uma partição da base de dados NUS-WIDE original contendo dez classes, por conta do volume de dados total da base, são elas: classes = {*birds*, *building*, *cars*, *cat*, *dog*, *fish*, *flowers*, *horses*, *mountain*, *plane*}. A mesma partição de dados foi adotada no trabalho de Shu et al. (2015).

A Tabela 2 apresenta uma descrição da partição dos dados da base NUS-WIDE. A

Figura 21 – Distribuição dos dados textuais da base NUS-WIDE completa

(a) Distribuição da quantidade de *Tags*

(b) Distribuição das ocorrências das *Tags*



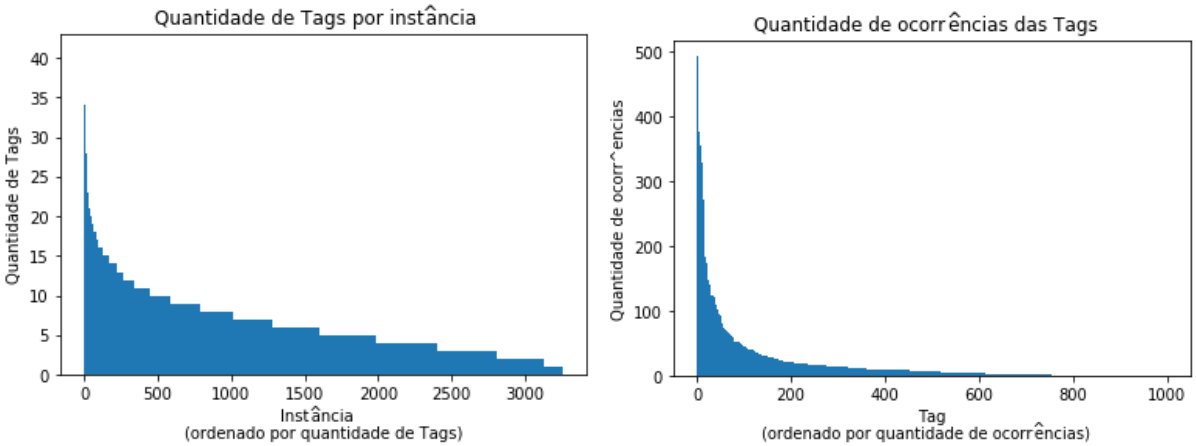
Fonte: Wanderley (2018)

partição é constituída por 3253 pares de imagens e textos descritivos, onde em média cada descrição apresenta 6,45 *Tags*. Em média, cada descrição apresenta 20,98 *Tags*, um valor expressivamente menor quando comparado com a base completa. A Figura 22 apresenta a distribuição da quantidade de *Tags* por descrição (22a) e a distribuição das ocorrências das *Tags* nas descrições (22b).

Figura 22 – Distribuição dos dados textuais da partição com 10 classes da base NUS-WIDE

(a) Distribuição da quantidade de *Tags*

(b) Distribuição das ocorrências das *Tags*



Fonte: Wanderley (2018)

A Tabela 3 apresenta a quantidade de instâncias por classe dentre a partição de 10 classes do NUS-WIDE.

Observando a distribuição das ocorrências das *Tags* nas descrições, representado na

Tabela 3 – Quantidade de instâncias por classe

Classe	**Quantidade de instâncias**
birds	371
building	349
cars	376
cat	333
dog	368
fish	382
flowers	336
horses	373
mountain	189
plane	176
Total	**3253**

Figura 22 (22b), percebe-se que a partir do vigésimo termo mais frequente, ao menos um dos 20 *Top Termos* está presente em mais de 98,0% das descrições das imagens. Uma conclusão que podemos tirar disso, é que quando o valor λ no algoritmo for maior ou igual a 20, observaremos em mais de 98,0% dos dados textuais um dos 20 termos mais frequentes.

5.2 Identificação dos Tópicos Semânticos

A primeira experimentação realizada tem como objetivo avaliar a capacidade de identificação dos tópicos semânticos a partir das características de alto nível extraídas das imagens. Neste experimento, é avaliada a acurácia dos classificadores binários construídos a partir das características visuais extraídas das imagens originais para cada problema semântico, de acordo com o Algoritmo 1, na página 46.

O classificador adotado baseia-se nas camadas finais de predição originais da VGG-16, realizando-se os devidos ajustes para um problema de classificação de dez classes (SIMONYAN; ZISSERMAN, 2014). Sendo assim, os classificadores selecionados são redes neurais na configuração de camadas totalmente conectadas contendo três camadas, sendo as duas primeiras com 4096 neurônios e com ativação ReLU e a última camada com 2 neurônios e com ativação *SoftMax*, para a classificação do tópico semântico.

O trenamento de cada classificador foi realizado aplicando o otimizador *Gradiente Descendente Estocástico* (GDE), utilizando uma taxa de aprendizagem decrescente iniciando em 10^{-5}, usando uma regra de parada antecipada, caso não haja reduções na função de perda maiores que 10^{-4} por quatro épocas consecutivas. O *Momentum* de Nesterov (NESTEROV, 1983) também foi adotado nos treinamentos. Esta configuração de treino

foi definida após experimentos preliminares de performance. Os experimentos foram executados segundo uma validação cruzada seguindo o método *k-fold* (KOHAVI, 1995), com $k = 10$.

O desempenho dos classificadores pode ser observado nos resultados apresentados na Tabela 4. Os classificadores desempenham uma tarefa binária em um problema balanceado (de acordo com o Algoritmo 1, na página 46), desse modo a acurácia base dos classificadores é definida em 0,5. Assim sendo, a Tabela 4 demonstra que a acurácia média dos classificadores define-se em torno de 0,67 (para os valores de λ avaliados) e com isso, entende-se que os classificadores não são aleatórios e que há de fato uma relação entre as características visuais e os tópicos semânticos considerados.

O desempenho dos classificadores também foram influenciados pela quantidade reduzida de instâncias de treino, conforme apresentado na Tabela 3 e de acordo com o Algoritmo 1, cada classificador foi treinado com aproximadamente 600 instâncias, *i.e.*, aproximadamente 300 exemplos positivos e aproximadamente 300 exemplos negativos aleatórios. Nos casos das classes "mountain" e "plane", a quantidade de exemplos positivos é ainda mais limitada.

Tabela 4 – Performance dos Classificadores de Tópicos Semânticos – Resultados

Quantidade de Classificadores (λ)	**Acurácia Média, Desv. Padr.**	**Melhor Acurácia**	**Pior Acurácia**
$\lambda = \mathbf{5}$	0,6748 ±0,0334	0,7908	0,5828
$\lambda = \mathbf{10}$	0,6830 ±0,0326	0,7944	0,5692
$\lambda = \mathbf{15}$	0,6743 ±0,0342	0,8430	0,4615
$\lambda = \mathbf{20}$	0,6791 ±0,0344	0,8153	0,4369
$\lambda = \mathbf{25}$	0,6678 ±0,0444	0,8343	0,4355
$\lambda = \mathbf{30}$	0,6614 ±0,0820	0,8523	0,4092

5.3 Reconstrução dos Dados Textuais

A segunda experimentação realizada busca avaliar a capacidade do modelo proposto em reconstruir os dados textuais. O experimento avalia a qualidade das *tags* aproximadas pelo modelo ($\hat{\mathbf{t}}$) comparando-as com as *tags* originais ($\mathbf{t}$).

As métricas utilizadas para a avaliação dos resultados neste experimentos foram as medidas de **precisão** e **revocação** além da **similaridade de Jaccard**. Todas apresentadas em detalhes no Apêndice B.

A Tabela 5 apresenta os resultados com a avaliação da qualidade dos textos gerados pelo modelo proposto. A implementação dos classificadores do modelo seguiu as mesmas configurações da seção anterior, bem como o restante da implementação do modelo como

um todo seguiu os detalhes mencionados no Capítulo 4. Similar à experimentação anterior, os experimentos foram executados segundo uma validação cruzada seguindo o método *k-fold* (KOHAVI, 1995), com $k = 10$.

Tabela 5 – Reconstrução dos Dados Textuais – Resultados

	Métrica	**Média**	**Desv. Padr.**
$\lambda = 5$	**Precisão**	0,2143	±0,0205
	Revocação	0,0957	±0,0082
	Jaccard	0,0735	±0,0041
$\lambda = 10$	**Precisão**	0,2661	±0,0257
	Revocação	0,1993	±0,0090
	Jaccard	0,1205	±0,0074
$\lambda = 15$	**Precisão**	0,2097	±0,0099
	Revocação	0,2343	±0,0137
	Jaccard	0,1164	±0,0034
$\lambda = 20$	**Precisão**	0,1259	±0,0031
	Revocação	0,3205	±0,0088
	Jaccard	0,0933	±0,0017
$\lambda = 25$	**Precisão**	0,1090	±0,0032
	Revocação	0,3442	±0,0109
	Jaccard	0,0850	±0,0022
$\lambda = 30$	**Precisão**	0,1460	±0,0063
	Revocação	0,3232	±0,0094
	Jaccard	0,1051	±0,0037

A partir dos resultados da Tabela 5, pode-se notar uma variação da precisão entre 10,0% e 26,0%, a depender do valor de λ. A precisão degrada-se para valores mais altos de λ, mas não de forma monotonicamente decrescente. A revocação varia entre valores de 9,0% até 34,0%, apresentando uma tendência de crescer a medida que o valor de λ cresce, provavelmente por conta da quantidade de classificadores recomendando mais termos. E por fim, a similaridade de Jaccard aponta que os grupos de termos (aproximados e originais) apresentam ainda uma fraca similaridade, com valores próximos aos 10,0%

O parâmetro λ implica no tamanho do dado textual aproximado $\hat{\mathbf{t}}$ e no número de classificadores no conjunto de classificadores $\mathbb{C}$. Quanto maior o valor de λ maior a quantidade de classificadores, e igualmente maior a quantidade de termos sugeridos pelo modelo. Sendo assim, elevados valores de λ também implicam e custos computacionais mais elevados. O impacto de mais termos sugeridos pelo modelo pode ser observado na Tabela 5 para os valores de revocação, ao passo que o valor de λ cresce, pode-se observar uma tendência do aumento dos valores de revocação, no entanto o valor de precisão não apresenta

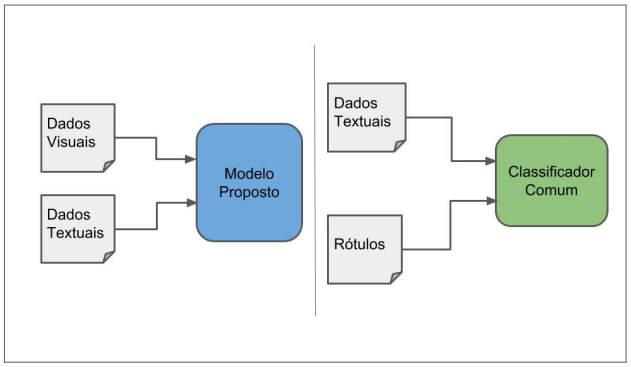
a mesma evolução, o que pode ser entendido como uma queda na qualidade dos termos recomendados para valores mais elevados de λ.

5.4 Classificação de Imagens

O terceiro experimento busca avaliar se a qualidade dos tópicos preditos pelo modelo proposto é suficiente para desempenhar uma tarefa de classificação de imagens. Para este propósito, o experimento definido consiste em um conjunto de tarefas de classificação, para cada uma das 10 classes mencionadas. Sendo assim, cada classe formula um problema de classificação binário. Os experimentos contrastam os resultados do modelo proposto com o desempenho das técnicas relacionadas mencionadas no Capítulo 3, aqui referenciadas como *linhas de base*. O Capítulo 3 apresenta em mais detalhes os métodos *linhas de base*, as quais são: HTL, TTI e DTN.

Reforçamos que a saída do modelo proposto são os dados textuais aproximados $\hat{\mathbf{t}}$ (como definido no Algoritmo 2), e que o modelo é treinado com os dados visuais e dados textuais associados, conforme o Algoritmo 1. Desse modo, para desempenhar uma atividade de classificação de imagens, um modelo auxiliar de classificação foi adotado. Este classificador (aqui referido como Classificador Comum) é, por sua vez, treinado com os dados textuais originais e com os respectivos rótulos das classes associadas para os problemas de classificação mencionados. A Figura 23 ilustra os componentes de treinamento do modelo proposto e do Classificador Comum.

Figura 23 – Diagrama de treino dos modelos para classificação de imagens



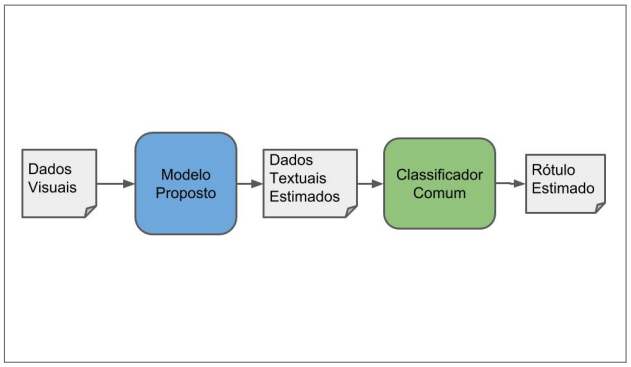
Fonte: Wanderley (2018)

Finalmente, para desempenhar a atividade de classificação, e assim avaliar a qualidade da informação aprendida pelo modelo proposto, o procedimento de classificação adotado nesta experimentação consiste em duas etapas. A primeira etapa aplica os dados visuais de uma nova instância de teste $\mathbf{x}$ ao modelo proposto treinado, o qual retorna dados textuais estimados $\hat{\mathbf{t}}$ para $\mathbf{x}$, conforme o Algoritmo 2. Em seguida, o vetor estimado $\hat{\mathbf{t}}$ é utilizado como entrada para o Classificador Comum, para que o rótulo y estimado pelo mesmo

seja então usado como o rótulo estimado para **x**. A Figura 24 ilustra o procedimento de classificação em duas etapas descrito.

Para compor o Classificador Comum, foi selecionada a técnica SVM) (CORTES; VAPNIK, 1995). Foram realizados experimentos preliminares para selecionar a melhor configuração de parâmetros para o classificador final, a combinação com o melhor desempenho foi escolhida para construir o classificador auxiliar para o experimento. Sendo assim, foram selecionados os parâmetros para uma SVM: núcleo (*kernel*) *Radial Basis Function* (Função de Base Radial) (RBF), $C = 1$ e $\gamma = 10^{-3}$, para treinar o classificador auxiliar.

Figura 24 – Diagrama do fluxo dos modelos para classificação de imagens



Fonte: Wanderley (2018)

Nesta experimentação, o problema consiste em uma classificação de imagens binária, onde a base de dados para um problema é composta por todas as imagens disponíveis para uma determinada classe (dentre as 10 classes selecionadas da base NUS-WIDE) e uma quantidade equivalente de exemplos negativos aleatórios extraídas das demais classes, para compor um problema balanceado de classificação. Diante disso, para este experimento foram formulados 10 problemas de classificação, os quais foram executados segundo uma validação cruzada seguindo o método *k-fold*, com $k = 10$. Os resultados estão reportados na Tabela 6, em termos da acurácia média ao longo da variação dos valores de λ, para cada problema de classificação particular, identificado pelo respectivo nome da classe (colunas). Além dos resultados do modelo proposto, estão os valores de desempenho dos métodos relacionados – *linhas de base*.

Os resultados foram bastante competitivos quando a solução proposta é comparada com as técnicas TTI e HTL. Para um tamanho de dimensionalidade suficiente, a acurácia em maioria foi melhor em comparação com essas *linhas de base*. Em comparação com a técnica DTN, por sua vez, a solução proposta foi menos competitiva, alcançando melhores resultados em 5 das 10 classes.

Similar ao observado no experimento anterior, elevados valores de λ (a partir de $\lambda = 15$) apresentaram uma melhor performance, mas não há uma garantia de que o desempenho será melhor para maiores valores de λ. Além do impacto em processamento para valores muito elevados de λ, no protótipo implementado, a regra utilizada para definir

Tabela 6 – Tarefa de Classificação de Imagens – Resultados em termos da acurácia média

	Linha de base			**Modelo Proposto (valor do parâmetro λ)**					
Classe	**DTN**	**TTI**	**HTL**	**5**	**10**	**15**	**20**	**25**	**30**
birds	**0,7800**	0,6850	0,7050	0,5380	0,6731	0,6716	0,6833	0,6878	0,7404
building	0,8000	0,6950	0,7100	0,5000	0,5413	0,5749	0,7391	**0,8415**	0,7794
cars	**0,8050**	0,7000	0,6750	0,6510	0,5770	0,5860	0,5980	0,6860	0,5940
cat	0,8250	0,7200	0,7100	0,5000	0,6364	**0,8311**	0,6537	0,7475	0,6982
dog	0,8050	0,7350	0,7000	0,7833	0,8275	**0,8294**	0,7777	0,7828	0,7780
fish	**0,7750**	0,6800	0,7100	0,6541	0,6008	0,5825	0,6510	0,6252	0,6255
flowers	0,7650	0,7010	0,7490	0,5000	0,6876	0,7430	0,7393	0,7172	**0,7747**
horses	0,7900	0,7050	0,7500	0,6317	0,7977	0,8211	0,7967	**0,8506**	0,7957
mountain	**0,8490**	0,7600	0,7600	0,5000	0,5000	0,5000	0,8463	0,6950	0,7795
plane	**0,8590**	0,8000	0,7700	0,5000	0,5000	0,6940	0,6374	0,6937	0,6427

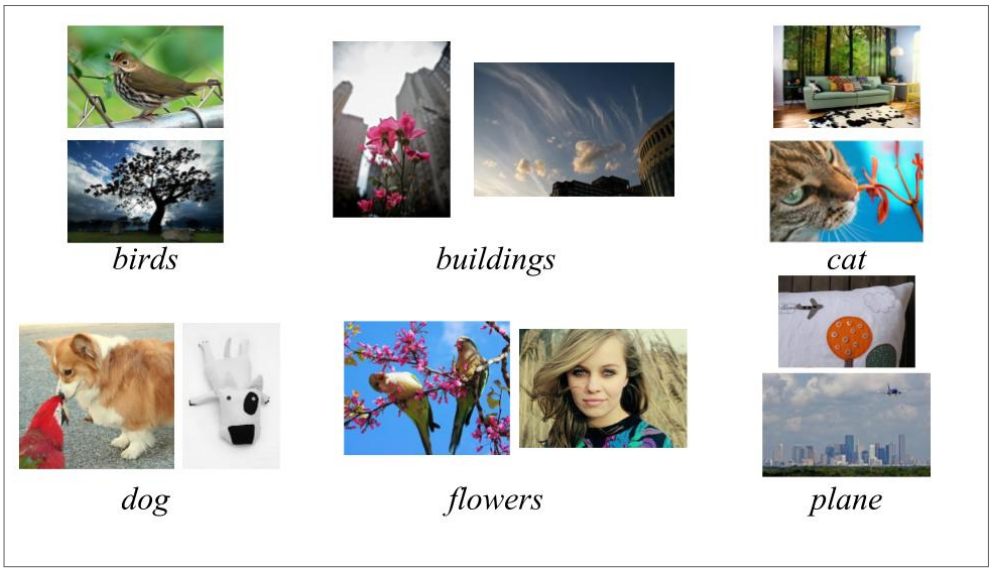
o espaço textual latente (discutida no Capítulo 4) tem grande influência no desempenho do modelo ao longo da variação do parâmetro λ, fazendo desta regra mais uma direção de melhorias deste trabalho.

Vale salientar que o desempenho pode ser melhorado a partir da adoção de outros classificadores mais recentes e robustos, até mesmo outra rede neural. Em nossos experimentos, foi adotada uma SVM com configurações padrão, pois o objetivo é avaliar a qualidade dos dados gerados pelo modelo proposto. Portanto, ainda há espaço para melhorias adotando classificadores mais fortes e realizando uma otimização de parâmetros mais exaustiva.

É importante lembrar que os resultados apresentados na tarefa de classificação de imagens são experimentos específicos para cada uma das dez classes consideradas, em conformidade com os experimentos conduzidos no trabalhos da linha de base (ZHANG et al., 2015). Não é o objetivo do trabalho treinar um único modelo genérico para desempenhar a tarefa de classificação para as dez classes simultaneamente. Por isso na Tabela 6 são apresentados os resultados individuais de desempenho do modelo proposto para diferentes valores do parâmetro λ, e não há uma análise geral considerando um resultado médio para todas as classes.

Os experimentos apresentados foram realizados na base de dados NUS-WIDE, que envolve imagens extraídas do *Flickr*. Existem algumas especificidades neste conjunto de dados que o torna difícil para problemas de classificação. As imagens são capturadas de várias maneiras, com resoluções variadas, além da existência de alguns ruídos visuais que comprometem a qualidade das imagens. Há também classes sobrepostas, por exemplo, edifícios e carros ou animais diferentes em uma mesma imagem, conforme exemplificado na Figura 25.

Figura 25 – Quadro com exemplos de sobreposição de classes na base NUS-WIDE



Fonte: Wanderley (2018)

5.5 Considerações Finais

Nesta seção final são feitas algumas considerações finais sobre os resultados obtidos nas três experimentações realizadas.

A primeira experimentação realizada buscou avaliar a capacidade de aprendizagem dos classificadores internos ao modelo na tarefa de identificar os tópicos semânticos de interesse. Os resultados apontaram que os classificadores alcançaram uma média geral de acurácia em torno de 67,0%, o que indica que os classificadores não são aleatórios (distantes do valor base de acurácia de 50,0%) e que, de fato, aprendem uma relação entre os tópicos semânticos e as imagens. No entanto, valores de acurácia mais altos não foram observados, provavelmente pelo fato de cada classificador ser treinado com uma quantidade reduzida de imagens, algo em torno de 600 exemplos.

A segunda experimentação avaliou a qualidade dos dados textuais sugeridos pelo modelo em relação aos dados textuais originais. Os resultados apontaram que com a variação dos valores de λ, para valores mais elevados, a precisão apresenta uma tendência de decrescimento, ao passo que a revocação apresenta uma tendência de crescimento. Isso pode ser justificado pelo fato que, quanto mais alto for o valor λ, mais termos são recomendados, pois existem mais classificadores sugerindo tópicos semânticos. Um ajuste nesse tipo de comportamento pode ser entendido como uma melhoria futura deste trabalho.

Por fim, a terceira experimentação realizada buscou avaliar a qualidade dos dados textuais retornado pelo modelo quando aplicados para uma tarefa de classificação de imagens, com a utilização de um classificador auxiliar. Os resultados obtidos foram confrontados com a performance de outras técnicas relacionadas. A arquitetura proposta se mostrou bastante competitiva, atingindo resultados superiores quando comparados com

os das técnicas TTI e HTL, e igualmente competitiva com relação à técnica DTN.

6 CONCLUSÕES

Neste capítulo final, apresentamos uma revisão das contribuições e atividades realizadas neste trabalho. Também são apresentadas direções para melhorias e evoluções do trabalho apresentado, diante algumas limitações encontradas.

6.1 Contribuições

Este trabalho apresentou uma proposta de arquitetura para resolver o problema de aprendizagem entre domínios heterogêneos, mais especificamente, aprendizagem entre domínios textuais e visuais. A técnica proposta tem como finalidade sugerir dados textuais para novas imagens de teste. Para o treinamento da arquitetura proposta são necessários pares de imagens e textos associados.

A arquitetura baseia-se em um conjunto de Redes Neurais Convolucionais específicas para definir a existência de determinados traços semânticos em imagens de teste. Ao longo do trabalho foi discutido o conceito de traços semânticos, como sendo abstrações de características que unem os dois domínios, isto é, características presentes em ambos os domínios. Para identificar os tópicos semânticos a partir dos dados textuais a arquitetura também envolve um grupo de modelos generativos, com a finalidade de identificar tópicos semânticos a partir de dados textuais e também de reconstruir dados textuais a partir dos dados semânticos, classificados pelo conjunto de classificadores.

Para avaliar a arquitetura proposta foi implantado um protótipo, para o qual foram realizados diversos experimentos com o intuito de avaliar a qualidade dos dados retornados pelo modelo proposto treinado.

Os resultados obtidos nos experimentos evidenciaram a capacidade de aprendizagem da arquitetura proposta, mesmo diante de dados relativamente difíceis, por exemplo, por conter sobreposição de classes em uma mesma imagem. Outra experimentação realizada buscou avaliar a qualidade dos dados textuais retornados pelo modelo proposto quando utilizados para realizar uma tarefa de classificação de imagens. Uma vez que a saída da arquitetura proposta são recomendações de dados textuais para imagens de teste, e não a classificação das imagens, um classificador auxiliar foi treinado para possibilitar essa experimentação. Os resultados obtidos foram confrontados com a performance de outras técnicas relacionadas. A arquitetura proposta se mostrou bastante competitiva, atingindo resultados superiores quando comparados com os das técnicas mais antigas e igualmente competitiva com relação às técnicas mais recentes.

O problema abordado neste trabalho é caracterizado pela dificuldade de relacionar domínios heterogêneos e assim, possibilitar uma aprendizagem conjunta entre domínios

distintos. A área da Transferência de Aprendizagem, em Aprendizagem de Máquina, busca tratar problemas de aprendizagem entre dados em contextos diferentes, inclusive o problema abordado neste trabalho.

Poucos trabalhos na literatura apresentam arquiteturas baseadas em Redes Neurais Profundas projetadas para desempenhar uma aprendizagem entre domínios textuais e visuais, inserindo a contribuição deste trabalho em uma área relevante e pouco explorada, conforme descrito na introdução do trabalho.

6.2 Trabalhos Futuros

Esta seção apresenta as limitações deste trabalho, bem como as direções para melhorias e continuidade desta pesquisa.

Os elementos presentes no protótipo implementado são entendidos como apenas uma instância da técnica proposta. Cabendo, portanto, a avaliação de diferentes formas de implementação da técnica. Mais precisamente, outros extratores de características visuais, classificadores de imagens e classificadores textuais podem ser explorados em novas experimentações. Igualmente, outras bases de dados podem ser avaliadas, além da NUS-WIDE.

A arquitetura proposta neste trabalho, necessita da definição de tópicos semânticos que possam ser derivados dos dados textuais originais. No capítulo onde o protótipo implementado é detalhado, uma regra para definição dos tópicos semânticos é apresentada, baseada nos termos mais frequentes do vocabulário dos dados textuais. A regra proposta é uma particularidade da implementação, e não limita o conceito mais amplo da arquitetura proposta. Sendo assim, uma direção de investigação para expandir este trabalho é a proposta e aplicação de diferentes regras para identificar tópicos semânticos. Por exemplo, regras alternativas baseadas em ontologias, para definir tópicos semânticos, podem ser exploradas.

Outro ponto relevante a ser evoluído neste trabalho é a capacidade de paralelismo da arquitetura proposta. Uma vez que a arquitetura apresentada baseia-se em um conjunto de classificadores independentes entre si, estes podem ser processados em paralelo. Muito embora esse seja mais um ponto positivo da arquitetura, a avaliação do paralelismo do modelo não foi realizada por limitações de *hardware*.

Finalmente, a arquitetura proposta pode ser evoluída para tratar problemas ainda mais complexos, como por exemplo, utilizar dados textuais na forma de textos em linguagem natural. Diferente dos dados textuais tratados ao longo deste trabalho, no formato de vetor de termos (*bag-of-words*). Indo mais além, a arquitetura proposta também pode inspirar o tratamento de outras combinações de domínios heterogêneos, como por exemplo, uma aprendizagem entre dados visuais (vídeo ou imagens) e dados de áudio.

REFERÊNCIAS

BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN 0387310738.

BLIER, L. *A BRIEF REPORT OF THE HEURITECH DEEP LEARNING MEETUP #5*. 2016. Disponível em: <https://blog.heuritech.com/2016/02/29/a-brief-report-of-the-heuritech-deep-learning-meetup-5/>.

CHUA, T.-S.; TANG, J.; HONG, R.; LI, H.; LUO, Z.; ZHENG, Y.-T. Nus-wide: A real-world web image database from national university of singapore. Santorini, Greece., 2009.

CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995.

DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, JSTOR, p. 1–38, 1977.

DONAHUE, J.; JIA, Y.; VINYALS, O.; HOFFMAN, J.; ZHANG, N.; TZENG, E.; DARRELL, T. Decaf: A deep convolutional activation feature for generic visual recognition. In: *International conference on machine learning*. [S.l.: s.n.], 2014. p. 647–655.

EVERINGHAM, M.; GOOL, L. V.; WILLIAMS, C. K. I.; WINN, J.; ZISSERMAN, A. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, v. 88, n. 2, p. 303–338, jun. 2010.

EVERINGHAM, M.; VANGOOL, L.; WILLIAMS, C. K. I.; WINN, J.; ZISSERMAN, A. *The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results*. 2012. Disponível em: <http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html>.

FEI-FEI, L.; FERGUS, R.; PERONA, P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding*, Elsevier, v. 106, n. 1, p. 59–70, 2007.

FLICKR; LUDICORP; YAHOO! *Flickr*. 2004. Disponível em: <https://www.flickr.com/>.

GLOROT, X.; BORDES, A.; BENGIO, Y. Deep sparse rectifier neural networks. In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. [S.l.: s.n.], 2011. p. 315–323.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A.; BENGIO, Y. *Deep learning*. [S.l.]: MIT press Cambridge, 2016. v. 1.

GRIFFIN, G.; HOLUB, A.; PERONA, P. Caltech-256 object category dataset. California Institute of Technology, 2007.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 770–778.

HEBB, D. O. *The Organizations of Behavior: a Neuropsychological Theory*. [S.l.]: Lawrence Erlbaum, 1949.

HINTON, G. E. Deep belief networks. *Scholarpedia*, v. 4, n. 5, p. 5947, 2009.

HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, National Acad Sciences, v. 79, n. 8, p. 2554–2558, 1982.

HOWARD, A. G.; ZHU, M.; CHEN, B.; KALENICHENKO, D.; WANG, W.; WEYAND, T.; ANDREETTO, M.; ADAM, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

IANDOLA, F. N.; HAN, S.; MOSKEWICZ, M. W.; ASHRAF, K.; DALLY, W. J.; KEUTZER, K. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and $<$ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: . [S.l.]: Morgan Kaufmann, 1995. p. 1137–1143.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2012. p. 1097–1105.

LECUN, Y.; BOSER, B.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W.; JACKEL, L. D. Backpropagation applied to handwritten zip code recognition. *Neural computation*, MIT Press, v. 1, n. 4, p. 541–551, 1989.

LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, IEEE, v. 86, n. 11, p. 2278–2324, 1998.

LECUN, Y. A.; BOTTOU, L.; ORR, G. B.; MÜLLER, K.-R. Efficient backprop. In: *Neural networks: Tricks of the trade*. [S.l.]: Springer, 2012. p. 9–48.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft coco: Common objects in context. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 740–755.

LU, J.; BEHBOOD, V.; HAO, P.; ZUO, H.; XUE, S.; ZHANG, G. Transfer learning using computational intelligence: a survey. *Knowledge-Based Systems*, Elsevier, v. 80, p. 14–23, 2015.

MCCALLUM, A.; NIGAM, K. et al. A comparison of event models for naive bayes text classification. In: CITESEER. *AAAI-98 workshop on learning for text categorization*. [S.l.], 1998. v. 752, n. 1, p. 41–48.

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

MINSKY, M.; PAPERT, S. *Perceptrons: An essay in computational geometry*. [S.l.]: Cambridge, MA: MIT Press, 1969.

NESTEROV, Y. E. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. *Dokl. Akad. Nauk SSSR*, v. 269, p. 543–547, 1983. Disponível em: <https://ci.nii.ac.jp/naid/10029946121/en/>.

NG, A.; NGIAM, J.; FOO, C. Y.; MAI, Y.; SUEN, C.; COATES, A.; MAAS, A.; HANNUN, A.; HUVAL, B.; WANG, T.; TANDON, S. *Deep Learning Tutorial*. Stanford University, 2016. Disponível em: <http://ufldl.stanford.edu/tutorial/>.

OQUAB, M.; BOTTOU, L.; LAPTEV, I.; SIVIC, J. Learning and transferring mid-level image representations using convolutional neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 1717–1724.

PALUMBO, L.; OGDEN, R.; MAKIN, A. D.; BERTAMINI, M. Examining visual complexity and its influence on perceived duration. *Journal of vision*, The Association for Research in Vision and Ophthalmology, v. 14, n. 14, p. 3–3, 2014.

PAN, S. J.; YANG, Q. et al. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, Institute of Electrical and Electronics Engineers, Inc., 345 E. 47 th St. NY NY 10017-2394 USA, v. 22, n. 10, p. 1345–1359, 2010.

QI, G.-J.; AGGARWAL, C.; HUANG, T. Towards semantic knowledge propagation from text corpus to web images. In: ACM. *Proceedings of the 20th international conference on World wide web*. [S.l.], 2011. p. 297–306.

QUATTONI, A.; TORRALBA, A. Recognizing indoor scenes. In: IEEE. *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. [S.l.], 2009. p. 413–420.

RAZAVIAN, A. S.; AZIZPOUR, H.; SULLIVAN, J.; CARLSSON, S. Cnn features off-the-shelf: an astounding baseline for recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. [S.l.: s.n.], 2014. p. 806–813.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group, v. 323, n. 6088, p. 533, 1986.

RUSSAKOVSKY, O.; DENG, J.; SU, H.; KRAUSE, J.; SATHEESH, S.; MA, S.; HUANG, Z.; KARPATHY, A.; KHOSLA, A.; BERNSTEIN, M. et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, Springer, v. 115, n. 3, p. 211–252, 2015.

SAENKO, K.; KULIS, B.; FRITZ, M.; DARRELL, T. Adapting visual category models to new domains. In: SPRINGER. *European conference on computer vision*. [S.l.], 2010. p. 213–226.

SHU, X.; QI, G.-J.; TANG, J.; WANG, J. Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation. In: ACM. *Proceedings of the 23rd ACM international conference on Multimedia*. [S.l.], 2015. p. 35–44.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. In: . [S.l.: s.n.], 2014.

SOEKHOE, D.; PUTTEN, P. van der; PLAAT, A. On the impact of data set size in transfer learning using deep neural networks. In: SPRINGER. *International Symposium on Intelligent Data Analysis*. [S.l.], 2016. p. 50–60.

SUTSKEVER, I.; HINTON, G. E.; TAYLOR, G. W. The recurrent temporal restricted boltzmann machine. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2009. p. 1601–1608.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 1–9.

VEEN, F. van. *The Neural Network Zoo*. The Asimov Institute, 2016. Disponível em: <http://www.asimovinstitute.org/neural-network-zoo/>.

WEISS, K.; KHOSHGOFTAAR, T. M.; WANG, D. A survey of transfer learning. *Journal of Big Data*, Nature Publishing Group, v. 3, n. 1, p. 9, 2016.

WELINDER, P.; BRANSON, S.; MITA, T.; WAH, C.; SCHROFF, F.; BELONGIE, S.; PERONA, P. Caltech-ucsd birds 200. California Institute of Technology, 2010.

WERBOS, P. Beyond regression:"new tools for prediction and analysis in the behavioral sciences. *Ph. D. dissertation, Harvard University*, 1974.

WIDROW, B.; HOFF, M. E. *Adaptive switching circuits*. [S.l.], 1960.

XIAO, J.; HAYS, J.; EHINGER, K. A.; OLIVA, A.; TORRALBA, A. Sun database: Large-scale scene recognition from abbey to zoo. In: IEEE. *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*. [S.l.], 2010. p. 3485–3492.

YOSINSKI, J.; CLUNE, J.; BENGIO, Y.; LIPSON, H. How transferable are features in deep neural networks? In: *Advances in neural information processing systems*. [S.l.: s.n.], 2014. p. 3320–3328.

ZEILER, M. D.; FERGUS, R. Visualizing and understanding convolutional networks. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 818–833.

ZHANG, X.; YU, F. X.; CHANG, S.-F.; WANG, S. Deep transfer network: Unsupervised domain adaptation. *arXiv preprint arXiv:1503.00591*, 2015.

ZHU, Y.; CHEN, Y.; LU, Z.; PAN, S. J.; XUE, G.-R.; YU, Y.; YANG, Q. Heterogeneous transfer learning for image classification. In: *AAAI*. [S.l.: s.n.], 2011.

APÊNDICE A – ARTIGO PUBLICADO

Contribuição científica resultante do trabalho apresentado. Artigo intitulado "*Transferring Knowledge From Texts to Images by Combining Deep Semantic Feature Descriptors*", publicado no evento "Conferência Internacional Unificada em Redes Neurais 2018" – "*The 2018 International Joint Conference on Neural Networks" (IJCNN 2018).*

DOI: 10.1109/IJCNN.2018.8489058
ISBN (eletrônico): 978-1-5090-6014-6
ISSN (eletrônico): 2161-4407

Transferring Knowledge From Texts to Images by Combining Deep Semantic Feature Descriptors

Miguel D. S. Wanderley and Ricardo B. C. Prudêncio
Centro de Informática (CIn)
Universidade Federal de Pernambuco (UFPE)
50670-901, Recife-PE, Brazil
{mdsw, rbcp}@cin.ufpe.br

***Abstract*— Deep learning techniques have been successfully applied to image processing tasks. Nevertheless, these techniques can be sensitive to the occurrence of small training sets, which is commonly faced when model training requires labelled instances. Transfer learning has been adopted to overcome this limitation by leveraging richer information in auxiliary domains to enhance the learning process in a target domain. In this paper, we proposed a new solution for transfer learning to label images (i.e., the target domain) by reusing labelled textual data (i.e., the auxiliary domain). A convolutional encoder is used to find latent features for images, while a probabilistic generative model is used to find semantic topics (traits) for texts. An ensemble of classifiers is then used to predict semantic topics to new input images according to their latent features. Experiments were performed to evaluate whether the latent features in both domains can actually be related and also to verify the use of the predicted semantic topics to classify images. Promising results were achieved compared to different baselines.**

I. Introduction

Recent deep learning techniques have been achieving expressive effectiveness in image related tasks, such as image labeling and classification [1], [2], [3], [4]. Some of the most successful models make use of Convolutional Neural Networks (CNN) such as AlexNet [1] and VGG [2]. In order to train these models, large labelled training sets are usually necessary. Nevertheless, in real applications, labelling data is commonly expensive and then only small training sets are available. In such cases, deep networks may be prone to overfitting and present poor performance [5]. In order to overcome this situation, transfer learning techniques can be adopted to learn deep models using auxiliary information [6].

Transfer Learning is a notably growing research topic in machine learning and computational intelligence. This topic investigates how to transfer information between different domains, whether this difference can be the data sources, modality, features spaces, data distributions or even tasks. In transfer learning, the main objective is to solve a learning task in a *target* domain by reusing information available in a *source* domain (which is somehow related to the target domain). Target domain usually presents poor information to build a good model, while the source domain provides relevant and possibly rich information, which can be leveraged or taken as a starting point for learning.

In the above context, different researchers have investigated how to reuse the features learned by deep networks. A common approach for transfer learning is to learn a deep model using a large source training dataset and then to fine-tune it using the target training data. More specifically, target data can be used to train a linear classifier on top of the last CNN layer trained on the source data. Zeiler *et al.* [7] demonstrated the benefits of this idea by pre-training a CNN on ImageNet, and then training a linear classifier on small image datasets in different contexts. Additionally, Yosinski *et al.* [8] trained an AlexNet on the ImageNet dataset as well and found that the first CNN layers act as generic and reusable feature filters, while the upper layers become specific to the classification task. Soekhoe *et al.* [9] found that for small target datasets, classification performance can be improved by just *freezing* the initially transferred layers (restore but not train) of the source domain network.

In the current work, we are focused on a more challenging scenario of transfer learning, which considers heterogeneous multimodal data, precisely texts and images. In fact, while labelled images can be scarce in a given target domain, textual data is easier to collect and label. Thus, improving image classification tasks by exploring correlated textual information is promising, and actually verified in previous work [10]. Since the availability of data is growing more and more, mainly in the wide web, there is an expressive amount of images and texts, usually correlated one to another (e.g., images associated to tags). These modalities are usually treated individually, with few works dealing with this sort of multimodal data, if compared to the overall growth of the machine learning field.

The algorithms proposed for heterogeneous transfer learning are based on various approaches, but the most recent and promising ones are based on deep learning techniques [10], [11], [5]. All of them propose algorithms which receive image and text data as training input for building a model that combines the multimodal data for a better performance. These mentioned works will be detailed in the following section.

In this work, we propose a new solution to the multimodal handling of images and textual data, in which an ensemble of classifiers is used to relate latent features respectively extracted for images and texts. Initially, a convolutional model is used to learn latent features for images and a generative model is used for extracting latent features for texts. We called each latent feature for texts as a *semantic*

trait or topic. A specific classification task is defined for each semantic topic in which the predictor attributes are the image latent features and the target is a binary label indicating if the image is related or not to the semantic topic. The training set for this classification task is produced from a paired corpus of images and related texts. The presented solution reduces the learning in the multimodal domain in a composition of multiple simpler decision problems by an ensemble of models.

The previous techniques intend to learn the relation between images and text data jointly. We propose a method in which the text data is taken as a set of semantic traits, and each semantic trait is learned separately, reducing the problem to a collection of less complex problems, i.e. binary problems. We propose a method based on a set of binary classifiers aimed to classify specific traits represented in the text domain, rather than learning in the more complex space of the complete text domain, as it is made in the mentioned jointly approaches.

The remainder of the paper is divided as follows. Section II presents an overview of transfer learning focused on deep learning and heterogeneous problems. Section III presents the proposed solution and its general architecture. Section IV presents the prototype implemented in a case study. Section V, in turn, presents the experiments and the obtained results. The final remarks are presented in Section VI.

II. Transfer Learning in Deep Networks

Generally speaking, transfer learning aims to optimize the performance of a learning system applied in a target domain by using auxiliary information available in a source domain [6]. There are different ways to categorize the previous work on transfer learning. Regarding feature spaces, transfer learning can be categorized as homogeneous or heterogeneous [12]. Homogeneous transfer learning is the scenario in which the feature space of both source and target domains is the same (e.g., source and target domains are related to images collected from different repositories but represented using the same descriptors). In heterogeneous transfer learning, in turn, the features used to represent instances in the source and target domain may be different. Heterogeneous transfer learning matches the multimodal image processing focused on our work.

Homogeneous transfer learning for deep networks was proposed, for instance, in [11]. In this paper, a deep network is built in such a way that the marginal and conditional distributions of source and target instances are matched. The marginal distribution of both target and source instances is computed by using the outputs of the network's hidden layers. Conditional distribution, in turn, is computed using final network output. In both cases, differences of distribution between source and target are computed using Maximum Mean Discrepancy (MMD), which is minimized during learning.

Although useful in some contexts, homogeneous transfer learning is limited in terms of the amount of information that can be transferred from different domains. There is a variety of previous work on heterogeneous transfer learning considering different learning approaches. For instance, in [10], collective matrix factorization is used to find a shared latent feature space between textual and image data. Tagged images are used to provide a bridge between the two domains. Reusing tagged images is a concept that we will adopt in our work. Nevertheless, we focus on transfer learning in deep networks.

In [5], a transformation mapping approach is proposed to learn latent representations for images and texts using Stacked Auto-Encoders. As in [10], a training set of paired text and images is adopted. The two SAEs are learned in such a way that their parameters are similar in the last layers. Additionally, the final latent representations of paired images and texts are similar too. Hence the SAEs served as a translator function which returned a shared latent representation of input images and texts if they are correlated. Image classification is performed based on the similarity (in the latent space) between each image given as input and the labelled text instances in a training set.

III. Semantic Feature Descriptors

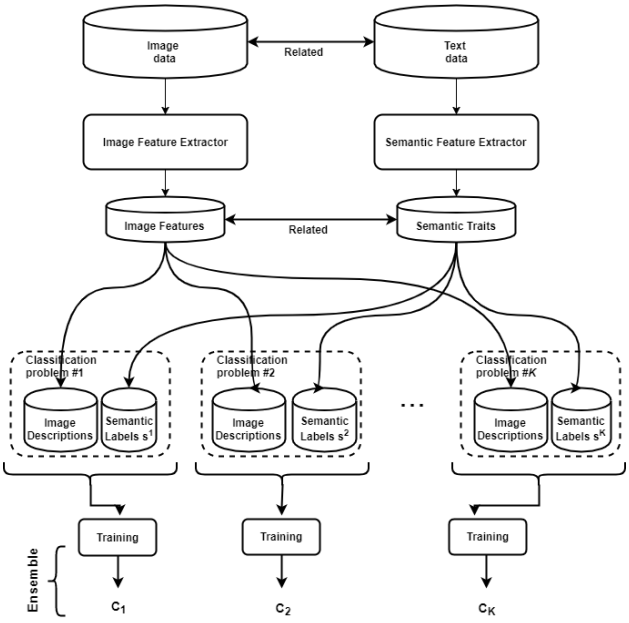


Fig. 1: Training process of the proposed solution.

In this section, we describe a new transfer learning solution for multimodal treatment of images. Two main assumptions are considered in this solution: (1) the first one is the presence of a set of paired images and texts in the target domain (e.g., tagged images), from which the semantic knowledge will be transferred from text to images; (2) additionally, a large set of images in an auxiliary domain is adopted to extract useful information for the target domain, where labelled images are scarce. More specifically, the auxiliary domain will be used to find latent feature descriptors which are reused to describe images in the target domain. The main objective in our work is to produce a new textual

representation for input images only based on the image information itself (i.e, for new input images in which no related textual information is available).

A key point in our work is the concept of a *semantic topic*. The occurrence of certain groups of words in a document may refer to a high-level topic or semantic trait, *e.g.*, the set of words {*airplane, aircraft, plane, aviation, jet, flying and fighter*} probably point to the same trait. Semantic topics will be derived from the corpus of documents available by using a feature extraction technique, as it will be seen.

The main idea in our solution is to predict semantic topics based on image features. In the proposed solution, an ensemble of image classifiers is adopted, each one specialized in a single specific semantic topic. The ensemble is trained by using the available set of paired images and related texts. Finally, the predictions returned by the ensemble for a given input image form a new representation of the image. This new representation can be used for different purposes like textual visualization, tagging, and classification.

The architecture for training the binary classifiers is presented in Figure 1. Both raw images and texts are transformed to reduced latent feature spaces. Each raw text document is represented originally by the presence of tags in a T-dimensional vocabulary $\mathbb{T}$, i.e., $\mathbf{d}_i = (d_i^1, \ldots, d_i^T)$, in which $d_i^j \in \{0, 1\}$ indicates that the j-th tag is presented in the i-th document, and $\mathbf{d}_i \in \mathbb{T}$. The raw representations will be transformed to another binary semantic feature vector $\mathbf{s}_i = (s_i^1, \ldots, s_i^K)$, in a feature space $\mathbb{S}$ ($\mathbf{s}_i \in \mathbb{S}$) in which $K << T$. Each new dimension $s_i^k \in \{0, 1\}$ indicates whether the text is related to the k-th semantic topic. Textual feature extraction is done in our work by a generative model Λ based on Bernoulli Naive Bayes, due to its capability of probabilistically dimension reduction. As this model returns probabilities for each semantic topic, we adopted a threshold for deriving the binary feature space. We represent the model as a function:

$$\mathbf{s} = \Lambda(\mathbf{d}) \tag{1}$$

As the case of textual documents, each raw image will be represented as a new latent vector $\mathbf{x}_i$. In this case, we adopted a Convolutional Feature Extractor, which has been successfully applied for providing rich representations of images [13]. We highlight that feature extraction from raw images is not the scope of this work. Convolutional Neural Networks have been shown excellent performance in previous work [8], [9] and are assumed as a good choice in the current work.

After feature extraction, a binary classification task is defined for each semantic topic. Training instances are produced by considering a set of n pairs of images and related texts $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^n$. The training set for the k-th task as defined as $\mathcal{D}_k = \{(\mathbf{x}_i, s_i^k)\}_{i=1}^n$, from which a classifier C_k is built. Any standard binary classification algorithm can be adopted in this step.

The trained model consists in a ensemble $\mathbb{C}$ of K classifiers for predicting the semantic topics based on the image latent features. The output of the ensemble for a given input image $\mathbf{x}$ is then a vector of predictions $\hat{\mathbf{s}} = (\hat{s}^1, \ldots, \hat{s}^K)$, in which $\hat{s}^k = C_k(\mathbf{x})$ (see Figure 2).

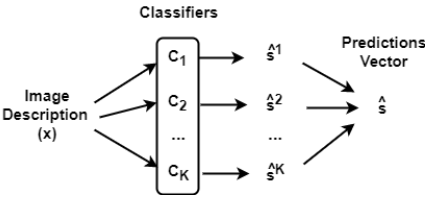


Fig. 2: Deployment of classifiers and final label prediction.

Algorithm 1 Training algorithm

Input:
$\mathcal{D} = \{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^n$: Set of paired images and texts
Output:
$\mathbb{C}$: Collection of semantic topics binary classifiers

1: **procedure** TRAIN
2: ***initialization***:
3: $\mathbb{C} \leftarrow \{\}$
4: ***for each*** *semantic topic* k ***do***:
5: $\mathbf{D}_k^+ \leftarrow \{(\mathbf{x}_i, s_i^k) | s_i^k = 1\}$ *//Positive images, i.e., for which related text contains the k-th semantic topic s^k.*
6: $\mathbf{D}_k^- \leftarrow \{(\mathbf{x}_i, s_i^k) | s_i^k = 0\}$ *//Negative images, i.e., for which related text does not contain the k-th semantic topic, limited to the same size of $\mathbf{D}_k^+$.*
7: $\mathbf{D}_k \leftarrow \mathbf{D}_k^+ \cup \mathbf{D}_k^-$
8: Train classifier C_k using $\mathbf{D}_k$
9: $\mathbb{C} \leftarrow \mathbb{C} \cup \{C_k\}$
10: ***end for***

IV. IMPLEMENTED PROTOTYPE

In order to evaluate the proposed solution, we implemented a prototype for image classification in the NUS-WIDE dataset [14]. NUS-WIDE is an open image set collected from Flickr, where each image has a related set of tags (words) as descriptors. In the implemented prototype, semantics topics are predicted for new input images only based on descriptors extracted by a convolutional model. Following the semantic topics are used as predicted attributes for image classification using an SVM as the classifier.

In the following subsections, we detail each component of the implemented prototype[1].

A. Image Feature Extractor

For image feature extraction, we adopted the VGG-16 model, which is a CNN used by the VGG team in

[1]Source code available at: *cin.ufpe.br/∼mdsw/ijcnn18*

Algorithm 2 Predicting algorithm

Input:

$\mathbf{x}$: Image features extracted from a new image to be labelled

Output:

$\hat{\mathbf{s}}$: Predicted semantic vector for the input image

1: **procedure** PREDICTION
2: ***initialization***:
3: $\quad \hat{\mathbf{s}} \leftarrow$ empty vector in the $\mathbb{S}$ space
4: ***for*** *Classifier* C_k *in* $\mathbb{C}$ ***do***:
5: $\quad \hat{s}^k \leftarrow C_k(\mathbf{x})$
6: ***end for***

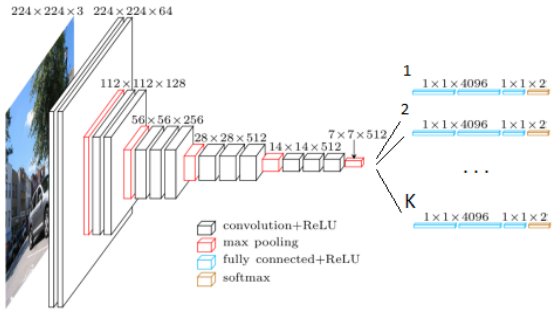


Fig. 3: Binary VGG classifiers. The initial 13 layers (trained using the ImageNet dataset) are used to feature extraction. The binary classifier is then augmented by 2 more hidden layers and the final output layer which will be adjusted for each specific semantic topic, *i.e.*, a binary classification problem. Figure adaptation of [16].

the ILSVRC-2014 competition with 16 layers [2]. The VGG weights were trained using an auxiliary dataset (the ImageNet[15] set). The feature extraction is performed in our work by the initial 13 layers. Other architectures and initial hidden layers could be used to feature extraction. This is an issue that will be better investigated in future work.

In the adopted VGG, the feature extraction process returns the image data in a reduced feature space, with the shape of 7 pixels width and height and 512 channels deep (7x7x512) (see Figure 3). This is the image feature space adopted by the binary semantic topic classifiers, which will be learned in isolation for each topic.

B. Text Feature Extractor

In the NUS-WIDE dataset, the original text inputs contain 1000 different tags to describe the images (i.e., represented in a feature space of dimensionality 1000). This representation is then reduced to a latent feature space (with dimensionality K) by the Λ generative model transformation.

Following the same principle of combining specific models for each semantic topic, the generative model Λ in this study is defined as a pool of generative models, one for each specific semantic topic. Let $\mathbb{G}$ be a pool of K generative models G, $\mathbb{G} = \{G_1, G_2, ..., G_k\}$, each G generative model is trained with a sub-space $\mathbb{T}_s$ of the initial text space $\mathbb{T}$, where $\mathbb{T}_s$ is the remaining text space $\mathbb{T}$ excluding the K top

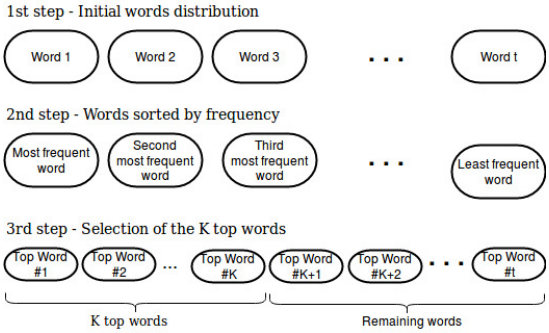


Fig. 4: Steps for composing of the text subspaces $\mathbb{S}$ (K top words) and $\mathbb{T}_s$ (Remaining words) based on the ranking of the most frequent words of $\mathbb{T}$.

words (words of highest occurrences, $\mathbb{S}$), thus $\mathbb{T} = \mathbb{T}_s \cup \mathbb{S}$. The objective of each generative model G_k is to predict the k-th top word (semantic topic) occurrence, for a given $\mathbf{d_s} \in \mathbb{T}_s$, where $\mathbb{T}_s = \{t_1, t_2, ..., t_s\}$, and also generate $\mathbf{d_s}$ for the given k semantic topic. By applying this text space reduction, the model equation (1) becomes:

$$\mathbf{s} = \Lambda(\mathbf{d_s}) \tag{2}$$

The text data $\hat{\mathbf{d}}$ may be reconstructed by the pool of generative models according to the equation (3), by combining the generated $\mathbf{d_s}$ value, from each G generative model in $\mathbb{G}$, for a given $\mathbf{s_i}$ and including the own latent features vector $\mathbf{s_i}$.

$$\hat{\mathbf{d}} = \mathbf{s_i} \cup \Lambda^{-1}(\mathbf{s_i}) = \mathbf{s_i} \cup \hat{\mathbf{d}}_\mathbf{s} \tag{3}$$

The Λ generative model was implemented using a pool of K Bernoulli Naive Bayes. The Figure 4 illustrates how the text space $\mathbb{T}$ is divided into the semantic topics $\mathbb{S}$ and the text subspace $\mathbb{T}_s$ based on the ranking of the most frequent words of $\mathbb{T}$.

C. Pool of Classifiers

For each semantic topic s^k, a single binary classifier is trained. As the classifier, we also adopted a VGG network, which is initialized with the feature extraction layers (13 layers) previously learned (i.e., these layers will be just transferred). The VGG is augmented with 3 layers that will be learned specifically for classifying the semantic topic: 2 hidden layers (as a dense layer) with 4096 neurons using a Relu activation and the output dense layer with 2 neurons activated by softmax, since it solves a binary classification problem. The final layers are learned by using the NUS-WIDE dataset.

The idea of reusing the initial convolutional layers trained with a larger dataset just to feature extraction was already performed for instance in [8] for homogeneous transfer learning (image to image). In our work, on the other hand, the idea is adopted for heterogeneous transfer learning (image to text), which was not yet investigated in the literature.

Each classifier is trained until convergence, avoiding overfitting by an early stop rule monitoring the minimum (10^{-3}) loss (binary cross entropy) in the validation set (20% of the training set). The training is done by the Stochastic Gradient Descent optimization algorithm, accelerated by the Nesterov method [17], with a learning rate of 10^{-5}, a learning decay of 10^{-6} and momentum of 0.9. These parameters were selected after some preliminary experiments.

V. Experiments and Results

In this section, we present the experiments performed, the implemented prototype and discuss the obtained results.

A. NUS-WIDE Data Set Description

Similar to [5], in our experiments a subset composed of 10 classes from the NUS-WIDE dataset [14]: {*birds, building, cars, cat, dog, fish, flowers, horses, mountain, plane*} was adopted, hereby called NUS-WIDE 10 Classes, with 3256 instances.

The image data in the original dataset has a varying value of width and height and 3 colors channels (RGB). A basic preprocessing is done for resizing all images to the fixed size of 224 width and 224 height with the same 3 color channels, resulting in a fixed (224 x 224 x 3) input data shape. No data augmentation is needed since the image feature extractor is already trained on the ImageNet dataset [15].

TABLE I: Dataset text data description

Dataset name	Number of instances	Number of tags per image			
		Max	Min	Average	Standard deviation
NUS-WIDE (10 classes)	3253	41	1	6.45	4.41

TABLE II: Words occurrences description

Dataset name	Number of top words	Words occurrences in descriptions			
		Max	Min	Average	Standard deviation
NUS-WIDE (10 classes)	1000	493	0	20.99	49.19

Table I presents a basic analysis of the text data in the dataset. Likewise, the Table II presents an analysis over the occurrences of the top words in the images descriptions in the dataset. The words occurrences distribution along the images descriptions can be seen in the Image 6, just as the Figure 5, plots the distribution of the words quantity in each image description set.

By observing the distribution of words into the descriptions, starting from the 20 most frequent words is enough to observe at least one top word in more than 98% of the descriptions text data instances.

B. Experiments - Semantic Topic Classification

In the experiments, we initially evaluated the pool of classifiers in order to verify whether the VGG image descriptors can be actually associated with the semantic topics.

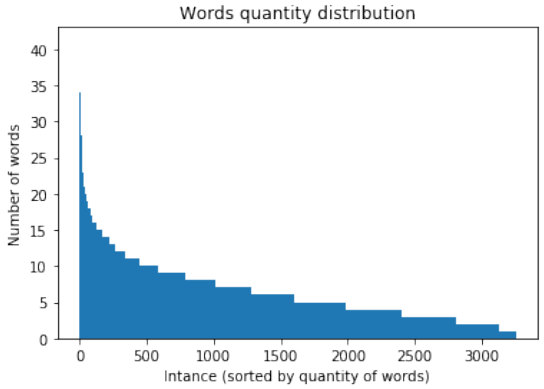


Fig. 5: Words distribution in the NUS-WIDE 10 Classes dataset

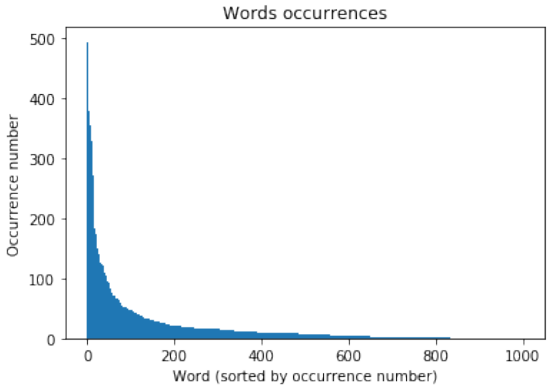


Fig. 6: Words occurrences in images descriptions in the NUS-WIDE 10 Classes dataset

For this purpose, each classifier in the pool was evaluated by adopting 10-fold cross validation over the NUS-WIDE dataset. The parameter K varied within the set $K = \{5, 10, 15, 20, 25, 30\}$. Performance is measured in terms of average accuracy across the classifiers (see Table Table III).

Average accuracy values were close to 0.67 regardless the parameter K. The default accuracy is 0.5, as the classes in the training sets are balanced. Thus, the obtained performances are nonrandom, and there actually is some relation between image descriptor and the learned semantic topics. The limited quantity of examples for training may result in some variability in the results, as it can be seen in the best and worst accuracy rates achieved. This variability reinforces the use of an ensemble strategy.

TABLE III: Classifiers performance

Number of Classifiers	Accuracy Mean, Std. Dev.	Best Accuracy	Worse Accuracy
5	0.6748, 0.0334	0.7908	0.5828
10	0.6830, 0.0326	0.7944	0.5692
15	0.6743, 0.0342	0.8430	0.4615
20	0.6791, 0.0344	0.8153	0.4369
25	0.6678, 0.0444	0.8343	0.4355
30	0.6614, 0.0820	0.8523	0.4092

C. Experiments - Image Classification

In the second experiment, one can verify whether the semantic topics suggested by the pool of classifiers are helpful to finally classify images. For this purpose, each category among the 10 classes of the dataset is treated as a binary label which will be predicted using the semantic topics. Thus, 10 binary image classification tasks are considered.

The parameter K implies in the dimensionality size of text representation of images and the number of classifiers in the pool. A lower value of K implies in a lower computational cost (since few classifiers are used), however, it may be not large enough to provide a good representation of semantic topics. Large K in turn may imply in performance gain in classification but leads to a higher computational cost.

In order to perform the binary classification tasks, for each class label, an SVM classifier with RBF kernel, $C = 1$ and $\gamma = 10^{-3}$ was adopted. We highlight that the predictor attributes for classification are only the semantic topics predicted by the pool of classifiers. The SVM is trained using images described by the semantic topics and associated with its class label. During the evaluation, the test images are initially transformed into predicted semantic topics by the classifier ensemble and then submitted to the SVM model for final classification. As in the previous experiment, we performed a 10-fold cross validation experiments, also varying the number of semantic topics adopted. Accuracy is also used as the performance measure.

In the Table IV, we present a comparison between the proposed solution and the baselines reported by Shu *et al.* in [5]: *weakly shared Deep Transfer Networks* (*weakly shared* DTN, presenting the best results between duftDTNs and sigDTNs [5]). Two other transfer algorithms involving text and image domains are also considered, *Heterogeneous Transfer Learning* (HTL) [10] and *Translator from Text to Image* (TTI) [18]. In our case, average accuracy was reported across the values of K.

- **Heterogeneous Transfer Learning for Image Classification (HTL):** Presents a matrix factorization based technique, where the relation between tagged images and the related tags is represented in a latent semantic space. A set of tagged images (co-occurrences of images and text tags) are used for learning a semantic view for each low-level image feature, i.e., train the model by a matrix factorization. A new target image instance is mapped into the semantic view, then a standard classifier can be built on the new image representation for a classification task [10].
- **Translator from Text to Images (TTI):** Presents a method to transfer knowledge across text domain and image domain for image classification task. The technique is based on a translator learned from the co-occurrences of text and images. The translator bridges the text and image domains into a latent *topic space* and is optimized by the proposed proximal gradient method. The translator propagates semantic labels from the text corpus to the images, in order to improve the image classification task [18].
- **Deep Transfer Networks (DTN):** Presents a deep neural network architecture built upon Stacked Auto-Encoders (SAE). The bottom part of the architecture is a pair of SAE, each one trained in each particular domain (text and image, in this case). the outputs of the SAEs are then combined and compose the input of the top part of the architecture, which is another SAE. The top SAE shares layers between the two first SAEs and builds a structure that encodes the two different domain inputs into a joint latent space representation. [11][5].

As presented in the Table IV, higher values of K lead to better classification performance in general. In fact, the best results (in boldface) were never observed for the lowest values of K (i.e., for $K = 5$ or 10). The observed increasing performance for higher values of K can be explained by the higher dimensionality of the latent text feature spaces. Nevertheless, as previously mentioned, there is a trade-off related to the computational cost as the K value increases, due to the need for training more classifiers. It is also important to mention that the proposed algorithm training can be easily parallelized since the internal classifier models are independent.

The results were very competitive when the proposed solution was compared to TTI and HTL. For a sufficient dimensionality size, accuracy was almost always better compared to these baselines. Compared to DTN in turn, which is the strongest baseline, the proposed solution was less competitive, achieving better results in 5 out of 10 classes. The adequate value of K, however, has to be found, which is an issue that will be more deeply investigated in future work. We also highlight that the performance can also be improved by adopting other classifiers. In our experiments, only an SVM with default parameters was adopted. Hence, there is still space for improvement by adopting stronger classifiers and by performing parameter optimization.

The presented experiments were performed in the NUS-WIDE dataset, which involves images extracted from Flickr. There are some particular difficulties in this dataset that turn this a difficult image set for classification. The images are captured in a wide range of ways, with varying sizes and some visual effects can be applied. There are also classes overlapping, for example, buildings and cars or different animals in the same image.

VI. Final Remarks

This paper presents a novel approach for transferring knowledge from text to images data. An ensemble of classifiers is used to predict semantic topics based on latent image features extracted by a deep model. The proposed architecture presented relevant improvements in labelling images tasks and achieved competitive results in image classification tasks, compared to different baselines.

Different future works can be pointed out. First, image feature extraction can be improved by investigating the adequate

TABLE IV: Baselines Mean Average Accuracy Comparison

		birds	**building**	**cars**	**cat**	**dog**	**fish**	**flowers**	**horses**	**mountain**	**plane**
Baselines	**DTN**	**0.7800**	0.8000	**0.8050**	0.8250	0.8050	**0.7750**	0.7650	0.7900	0.8490	**0.8590**
	TTI	0.6850	0.6950	0.7000	0.7200	0.735	0.6800	0.7010	0.7050	0.7600	0.8000
	HTL	0.7050	0.7100	0.6750	0.7100	0.7	0.7100	0.7490	0.7500	0.7600	0.7700
Proposed	**k=5**	0.5380	0.5000	0.6510	0.5000	0.7833	0.6541	0.5000	0.6317	0.5000	0.5000
	k=10	0.6731	0.5413	0.5770	0.6364	0.8275	0.6008	0.6876	0.7977	0.5000	0.5000
	k=15	0.6716	0.5749	0.5860	**0.8311**	**0.8294**	0.5825	0.7430	0.8211	0.5000	0.6940
	k=20	0.6833	0.7391	0.5980	0.6537	0.7777	0.6510	0.7393	0.7967	**0.8463**	0.6374
	k=25	0.6878	**0.8415**	0.6860	0.7475	0.7828	0.6252	0.7172	**0.8506**	0.6950	0.6937
	k=30	0.7404	0.7794	0.5940	0.6982	0.7780	0.6255	**0.7747**	0.7957	0.7795	0.6427

dimensionality size. In our work, a fixed VGG-16 architecture was used for feature extraction. Other convolutional models can be evaluated too. Similarly, other generative models can be adopted, for example, graphical models like Restricted Boltzmann Machine and Deep Belief Networks due to the recent good performance of these techniques in some generative tasks.

Finally, the semantic feature spaces produced by the proposed solution can provide useful information for other tasks like image tagging and image captioning. New experiments in other contexts and tasks can be performed in the future, and also including different datasets. The proposed meta-architecture could also be explored in different heterogeneous domains combinations, for example, audio, natural language, biology, finance, business, and others.

REFERENCES

[1] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks". In Advances in Neural Information Processing Systems 25, Harrahs and Harveys, 2012.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition". In International Conference on Learning Representations (ICLR), San Diego, 2015.

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going Deeper with Convolutions". In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, 2015.

[4] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition". In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[5] X. Shu, G. Qi, J.Tang and J. Wang, "Weakly-Shared Deep Transfer Networks for Heterogeneous-Domain Knowledge Propagation". In Proceedings of the 23rd ACM international conference on Multimedia - MM '15 p. 35-44, 2015.

[6] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue and G. Zhang, "Transfer learning using computational intelligence: A survey". IEEE Transactions on Knowledge and Data Engineering, May 2015.

[7] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks". Computer vision (ECCV), pp. 818-833, 2014.

[8] J. Yosinski, J. Clune, Y. Bengio and H. Lipson, "How transferable are features in deep neural networks?. In Advances in Neural Information Processing Systems 27 (NIPS 2014), Montral, 2014.

[9] D. Soekhoe, P. van der Putten and A. Plaat, "On the Impact of data set Size in Transfer Learning using Deep Neural Networks". In The 15th International Symposium on Intelligent Data Analysis, Stockholm, 2016.

[10] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G.-R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification". In AAAI , 2011.

[11] X. Zhang, F. X. Yu, S.-F. Chang, and S. Wang, "Deep transfer network: Unsupervised domain adaptation". arXiv , 2015.

[12] K. Weiss, T. M. Khoshgoftaar and D. Wang, "A survey of transfer learning". Journal of Big Data, 3:9, May 2016.

[13] M. Oquab, L. Bottou, I. Laptev e J. Sivic, "Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks". IEEE Conference on Computer Vision and Pattern Recognition, Columbus, 2014.

[14] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "NUS-WIDE: A Real-World Web Image Database from National University of Singapore". ACM International Conference on Image and Video Retrieval. Greece. Jul. 8-10, 2009.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database". In CVPR , pages 248255, 2009.

[16] L. Blier, "A BRIEF REPORT OF THE HEURITECH DEEP LEARNING MEETUP #5", WordPress.com, 29 February 2016. [Online]. Available: "https://blog.heuritech.com/2016/02/29/a-brief-report-of-the-heuritech-deep-learning-meetup-5/". last access in April 2018.

[17] Nesterov, Y., "A method for unconstrained convex minimization problem with the rate of convergence o(1/k2)". Doklady ANSSSR (translated as Soviet.Math.Docl.), vol. 269, pp. 543 547, 1983.

[18] G-J Qi, C. Aggarwal and T. Huang, "Towards semantic knowledge propagation from text corpus to web images". WWW '11 Proceedings of the 20th international conference on World wide web, Pages 297-306, 2011.

APÊNDICE B – MÉTRICAS DE AVALIAÇÃO

Neste apêndice são apresentadas as descrições das métricas de avaliação adotadas para a elaboração dos experimentos apresentados neste trabalho.

As métricas utilizadas para a avaliação dos resultados apresentados neste trabalho foram as medidas de **precisão** e **revocação** além da **similaridade de Jaccard**.

A Equação B.1 formaliza a métrica **precisão**, como sendo, no contexto da aplicação deste trabalho, uma relação entre todos os termos realmente presentes no dado textual original dentre os termos sugeridos. Já a métrica **revocação** pode ser entendida como a relação entre todos os termos realmente presentes no dado textual original recomendados dentre todos os termos realmente presentes no dado textual original, como pode ser visto na Equação B.2. A Figura 26 ilustra a definição das métricas precisão e revocação.

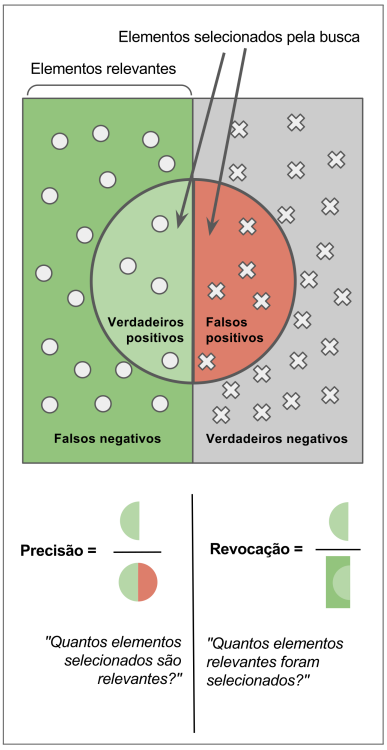
$$\text{Precisão} = \frac{|\{\text{termos relevantes}\} \bigcap \{\text{termos recuperados}\}|}{|\{\text{termos recuperados}\}|} \tag{B.1}$$

$$\text{Revocação} = \frac{|\{\text{termos relevantes}\} \bigcap \{\text{termos recuperados}\}|}{|\{\text{termos relevantes}\}|} \tag{B.2}$$

Por fim, a **similaridade de Jaccard** busca indicar o quanto dois conjuntos são similares, considerando os seus elementos componentes. No nosso caso, os dois conjuntos são os dados textuais originais e os dados textuais recomendados pelo modelo. A Equação B.3 formaliza a similaridade de Jaccard, $J(A, B)$, entre dois conjuntos A e B como sendo a relação entre a intersecção entre A e B e a união entre A e B:

$$J(A, B) = \frac{|A \bigcap B|}{|A \bigcup B|} = \frac{|A \bigcap B|}{|A| + |B| + |A \bigcup B|} \tag{B.3}$$

Figura 26 – Ilustração das Métricas Precisão e Revocação



Fonte: Wanderley (2018)