Spring 4-2021

Data-Efficient Machine Learning with Focus on Transfer Learning

Shuteng Niu
Embry-Riddle Aeronautical University

Data-Efficient Machine Learning with Focus on Transfer Learning

by

Shuteng Niu

A dissertation submitted to the Faculty of Embry-Riddle Aeronautical University in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Electrical Engineering and Computer Science

Embry-Riddle Aeronautical University
Daytona Beach, Florida

April 2021

Data-Efficient Machine Learning with Focus on Transfer Learning

by Shuteng Niu

This dissertation was prepared under the direction of the candidate's Dissertation Committee Chair, Dr. Houbing Song, and has been approved by the members of the dissertation committee. It was submitted to the College of Engineering and accepted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in Electrical Engineering and Computer Science.

Houbing Song

Houbing Song, Ph.D.
Committee Chair

Radu F. Babiceanu, Ph.D.
Committee Member

Richard S. Stansbury
Digitally signed by Richard S. Stansbury
Date: 2021.04.22 09:55:31 -04'00'

Richard S. Stansbury, Ph.D.
Committee Member

Tianyu Yang
Digitally signed by Tianyu Yang
DN: cn=Tianyu Yang, o=ERAU, ou, email=yang482@erau.edu, c=US
Date: 2021.04.22 15:44:17 -04'00'

Tianyu (Thomas) Yang, Ph.D.
Committee Member

Xingquan (Hill) Zhu
Digitally signed by Xingquan (Hill) Zhu
Date: 2021.04.22 14:51:56 -04'00'

Xingquan (Hill) Zhu, Ph.D.
Committee Member

Timothy A Wilson

Timothy A. Wilson
2021.04.23 12:27:39 -04'00'

Timothy A. Wilson, Sc.D. — Date
Chair, Electrical Engineering and Computer Science

Digitally signed by Maj Dean Mirmirani
DN: cn=Maj Dean Mirmirani, o=Embry-Riddle Aeronautical University, ou, email=mirmiram@erau.edu, c=US
Date: 2021.04.26 12:25:22 -04'00'

Maj Mirmirani, Ph.D. — Date
Dean, College of Engineering

April 26, 2021

Lon Moller, J.D. — Date
Senior Vice President for Academic Affairs and Provost

“A man is rich in proportion to the number of things which he can afford to let alone.”

Henry David Thoreau

Abstract

Machine learning (ML) has attracted a significant amount of attention from the artificial intelligence community. ML has shown state-of-art performance in various fields, such as signal processing, healthcare system, and natural language processing (NLP). However, most conventional ML algorithms suffer from three significant difficulties: 1) insufficient high-quality training data, 2) costly training process, and 3) domain discrepancy. Therefore, it is important to develop solutions for these problems, so the future of ML will be more sustainable. Recently, a new concept, data-efficient machine learning (DEML), has been proposed to deal with the current bottlenecks of ML. Moreover, transfer learning (TL) has been considered as an effective solution to address the three shortcomings of conventional ML. Furthermore, TL is one of the most active areas in the DEML. Over the past ten years, significant progress has been made in TL.

In this dissertation, I propose to address the three problems by developing a software-oriented framework and TL algorithms. Firstly, I introduce a DEML framework and a evaluation system. Moreover, I present two novel TL algorithms and applications on real-world problems. Furthermore, I will first present the first well-defined DEML framework and introduce how it can address the challenges in ML. After that, I will give an updated overview of the state-of-the-art and open challenges in the TL. I will then introduce two novel algorithms for two of the most challenging TL topics: distant domain TL and cross-modality TL (image-text). A detailed algorithm introduction and preliminary results on real-world applications (Covid-19 diagnosis and image classification) will be presented. Then, I will discuss the current trends in TL algorithms and real-world applications. Lastly, I will present the conclusion and future research directions.

Acknowledgements

First of all, I would like to thank my Ph.D. advisor, Dr. Houbing Song, for supporting me during the past four years. Dr. Song is a professional full of knowledge and wisdom, and he is one of the smartest people I know. He is a great advisor who always leads me with patience and a good and warm friend. I am very fortunate to have worked with Dr. Song, and I hope that I could be as lively, enthusiastic, and energetic as him and someday be able to flourish in academia works as he does. Moreover, I want to thank my secondary advisor, Dr. Jianhua Liu, for providing me with wise advice and guidance during my Ph.D. time.

I want to thank all my committee members for years of guidance and collaboration in my Ph.D. life. I am grateful that Dr. Radu Babiceanu provided me kind help with my career and his fantastic mentoring of system engineering-related topics. Great thanks to Dr. Tianyu Yang for his excellent teaching in several core classes and the Air-Force proposal writing collaboration. Moreover, Dr. Yang gave generous support for my future career. I also want to thank Dr. Richard Stansbury and Dr. Xingquan Zhu for their valuable inputs to my research in machine learning and big data. Moreover, Dr. Stansbury and Dr. Zhu supported me with wisdom advice for my future career.

I also need to express my gratitude to all the SONG Lab members who gave me helps in academia. Especially, I appreciate Jian Wang, Yushan Jiang, Kai Zhang and Bowen Chen for their collaboration on many machine learning projects and algorithm development. Moreover, I want to thank Dr. Xianping Du and Yihao Hu for the inspiring discussions we had. Many Thanks to Dr. Timothy Wilson for the advice to my faculty position interviews. Furthermore, I also want to thank my Ph.D. financial support agents: the Center for Advanced Transportation Mobility (CATM), USDOT

Grant #69A3551747125 and through the National Science Foundation under Grant No. 1956193.

Finally and most importantly, I want to thank my father, Lianfu Niu, my mother, Yanxiu Su, for their selfless support so that I can always pursue the path that I love. I also want to thank my parents for always warming my heart with unconditional love and standing by my side when troubles come. Moreover, I also want to thank all other families who always believe in me and support me. I want to dedicate this milestone of my life to them for their love and support.

Contents

List of Figures

List of Tables

For my parents

Chapter 1

Introduction

1.1 Background and Motivation

Machine Learning (ML) was proposed decades ago as a sub-field of artificial intelligence, and it is now attracting more and more attention. In ML history, there were two major winters in 1974 - 1980 and 1987 - 1993. ML was not preferred by academia and the industry during the previous winter due to its unstable performance and limited computational power. To break the winter, many powerful processors such as graphics processing unit (GPU) and Tensor Processing Unit (TPU) were built to improve the performance of deep ML models [10, 11]. After that, with the evolution of the internet, collecting massive training data became much easier. Recently, With these two improvements, the modern ML (deep learning) has been successfully applied to various areas in our daily life, such as smartphones, health care, and smart cities.

In the past decade, the academia and the industry have achieved significant breakthroughs in several ML disciplines, such as supervised learning, semi-supervised learning, unsupervised learning, and reinforcement learning. Moreover, supervised learning and reinforcement learning have led the ML trend with their superb and robust performances. Generally, supervised learning requires a massive amount of well-labeled data and computational power for the training process, which is not always feasible to many users. Reinforcement learning is very computationally expensive due to its unstable training process (non-convergence). Therefore, data-efficient machine learning (DEML) has been proposed to create a more sustainable future for modern ML. DEML is a concept that covers all the techniques to address the incompatible training data and the computational power. With the help of DEML, the modern ML can become more efficient in training and more robust in prediction. Furthermore, as a subfiled in DEML, transfer learning (TL) has been attracting more and more attention since it can effectively deal with the shortcomings of supervised learning and reinforcement learning [12].

Unlike other ML disciplines, the inspiration of TL is closely related to bionics. It mimics humans' ability to generalize knowledge from one area to another similar area. For example, English speakers usually can learn Spanish with less effort because English and Spanish share many common rules in pronunciation and grammar. With transferring the common knowledge, one does not need to start learning Spanish from scratch. Similarly, TL aims to use the least effort to develop a target model by transferring knowledge stored in other models. This concept has greatly expanded the use of ML to many performance-critical areas, such as biomedical informatics, NLP, and smart-and-connected system. However, traditional TL only tends to transfer information among similar domains and tasks. Negative transfer occurs when there is a larger domain discrepancy. In the future, we hope to develop more powerful TL algorithms that fit for distant domain transfer and the cross-modality transfer.

1.2 Challenges

This dissertation discusses major challenges in modern ML and TL. Firstly, deep learning is the mainstream in modern ML. Deep learning is facing three major challenges [13]:

- Insufficient training data causes significant performance degradation.
- Advance computational power is not accessible to everyone and We are reaching computational limits for deep learning.
- The modern deep learning is not computationally expensive and data-dependent by accident, but by design.

Primarily, most deep learning algorithms require a massive amount of training data. However, this condition does not hold in many real-world problems. Moreover, collecting and manually labeling a massive data set is too costly to do, and artificial data lacks distribution diversity. Furthermore, the incompatible computational power is another adversity of modern ML. Recently, there are several potential solutions to align the model depth and the computational power, such as cloud computing [14], edge computing [15], and parallel computing [16]. However, these solutions are not always reliable due to poor communication stability and security. In addition, the computational power is reaching the limit for deep learning. Moreover, most ML algorithms assume that the training data and the testing data are independent distributed (i.i.d.), but it does not stand for most practical ML problems. As a consequence, insufficient training data lead to a large distribution mismatch between the training and the testing data. The distribution mismatch can result a great performance decrease when a

ML model is applied to real-world applications. Moreover, most ML algorithms rely on massive training data due the architecture designs.

In traditional TL, there are major challenges:

- it assumes that the source domain and the target domain are closely related to each other.
- it cannot transfer knowledge between different modalities.
- it focuses on transferring on traditional and statistical models.

It has been proved that transfer learning is able to handle two critical machine learning problems: 1) insufficient training data, and 2) domain distribution mismatch. Theoretically, transfer learning algorithms aim to develop robust target models by using only a small set of target training data and transferring knowledge learned from other domains and tasks. Previously, the concept of adaptation layer with domain distance measurements was first proposed by [17]. It allows us to transfer knowledge between deep neural networks. In general, conventional transfer learning algorithms assume that the source domains and the targets share a certain amount of common information. However, this assumption does not always hold in many real-world applications, such as medical image processing [18, 19], rare species detection [20] and recommendation systems [21, 22]. In addition, transferring between two loosely related domains usually causes negative transfer [23–25], meaning that the knowledge transfer starts hurting the performance on the task in the target domain, and produces worse performance than non-transfer models.

As these problems have become new challenges, this dissertation proposes to develop reliable solutions: 1) DEML framework and 2) TL.

1.3 Proposed Methodology

In this dissertation, the author proposes the following methodologies:

- DEML Framework and Evaluation System
- A Decade Survey of Transfer Learning
- Conventional Transfer Learning for Solid Waste Sorting
- Feature-based Distant Domain Transfer Learning
- Distant Domain Transfer Learning for Medical Imaging
- Cross-Modality Transfer Learning for Image-Text Information Management

1.4 Contribution

To distinguish this dissertation from other studies, this dissertation pays attention to several important but not-well investigated problems, such as DEML, DDTL. and CMTL. Moreover, in this dissertation, there are four major contributions: 1) the author proposes the concept of DEML and develops well-defined framework with a evaluation system, 2) the author conducts the most recent TL literature review that covers novel topics (DDTL and CMTL), 3) the author introduces two novel algorithms to deal with two most challenging TL problems: DDTL and CMTL, and 4) the author also presents two real-world applications with TL in this dissertation.

1.5 Organization of the Dissertation

Finally, the remainder of this dissertation is structured as follows: In section-2, the author will give a comprehensive review of TL in the past decade, and this overview can help professional to find well-suited methods for different situations quickly. And then, Section-3 will introduce a novel DEML framework and a evaluation system. Moreover, the author will introduce a TL method for solid waste sorting. Next, a novel distant domain TL algorithm will be discussed in Section-4. After that, the author will demonstrate a DDTL application on a medical imaging task. Moreover, a corss-modality TL algorithm will be introduced in Section-5. Finally, a conclusion of the dissertation and a discussion of the future directions will given in Section-6.

Chapter 2

A Decade Survey of Transfer Learning (2010 - 2020)

Transfer learning (TL) has been successfully applied to many real-world problems that traditional machine learning (ML) cannot handle, such as image processing, speech recognition, and natural language processing (NLP). Commonly, TL tends to address three main problems of traditional machine learning: (1) insufficient labeled data, (2) incompatible computation power, and (3) distribution mismatch. In general, TL can be organized into four categories: transductive learning, inductive learning, unsupervised learning, and negative learning. Furthermore, each category can be organized into four learning types: learning on instances, learning on features, learning on parameters, and learning on relations. This article presents a comprehensive survey on TL. In addition, this chapter presents the state of the art, current trends, applications, and open challenges.

Transfer learning (TL) has attracted a significant amount of attention from the artificial intelligence community. TL can effectively handle challenging machine learning problems, such as lack of sufficient training data and changes in the concepts being learnt. Over the past 10 years, significant progress has been made in TL. The author presents an updated survey by demonstrating the state-of-the-art, current trends, and open challenges in the field. While most recent surveys equally cover mainstream topic on TL, our survey extends that by identifying and discussing the most challenging TL problems, such as distant domain and cross-modality TL. The survey promotes the positive applications of transfer learning to foster a broader community in the field.

2.1 Introduction

Recently, ML has made breakthroughs in a number of different fields, including but not limited to image processing, speech recognition, and natural language processing (NLP). With state-of-the-art performances, ML techniques have been applied to more and more real-world problems that traditional statistical learning methods cannot handle.

Commonly, traditional ML relies on a massive amount of training data. It assumes one critical condition: the training data and the testing data are drawn from the exact same distribution. However, this assumption does not always hold in many real-world problems. As such, most conventional ML algorithms usually suffer from three main difficulties: insufficient data, incompatible computation power, and distribution mismatch. First of all, various solutions have been proposed to address the first two problems, such as data argumentation, data synthesis, distributed learning, and cloud computing. However, each of these proposed solutions suffers from some adversities, such as regarding cost, efficiency, and security. Recently, transfer learning (TL) has been brought to our attention to deal with all three difficulties.

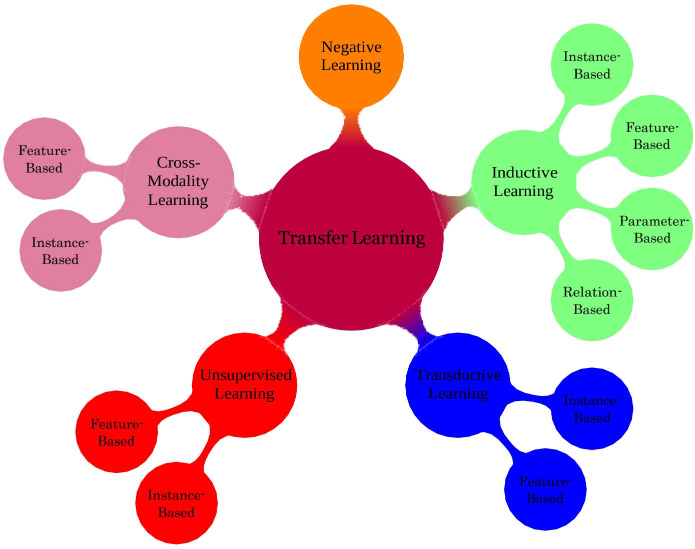


Figure 2.1: Mindmap of Transfer Learning

Primarily, TL aims to solve the target task by leveraging the knowledge learned from source tasks in different domains, so it does not need to learn from scratch with a massive amount of data [23, 26, 27]. As such, TL first can address the most significant issue, insufficient well-labeled training data. Moreover, the time and computation resources required for training a model can also be greatly decreased since pre-learned knowledge from other domains and tasks can be reused. Furthermore, the distribution mismatch can cause significant performance degradation on ML models. TL can also address it by fusing knowledge from one or multiple different domains.

In this chapter, the most representative works on TL in the past decade will be introduced and organized into different categories. Firstly, the author categorizes TL methods into two levels. As shown in Figure-2.1, in the first level, according to the availability of well-labeled data and the data modality in the source and target domains, it is categorized into five sub-fields: inductive TL, transductive TL, cross-modality TL, unsupervised TL, and negative TL respectively. Innovatively, each sub-field in the first level is again categorized into four different learning types: learning on instances, learning on features, learning on parameters, and learning on relations. Moreover, many successful real-world TL applications will also be introduced to emphasize TL's importance to the industry. And more, negative learning also plays a vital role in TL, which is an essential topic of TL but lacks attention. It is not studied by different learning types in the second level. In stead, it is discussed from two perspectives: problem definition and algorithms. In this survey, a number of state-of-the-art works on negative transfer will be discussed. Furthermore, open challenges and future research directions are also discussed in this survey.

Comparing with other recent surveys on TL, as shown in Table-2.1, the author makes several main improvements and contributions in this review. The following outlines the main contributions of our survey:

Table 2.1: Comparison of Recent Surveys on TL

	Statistical	Deep Learning	Homogeneous	Heterogeneous	Negative	Cross-Modality	Applications
[23]	Yes	No	Yes	No	Yes	No	Yes
[28]	Yes	Yes	Yes	Yes	Yes	No	Yes
[29]	Yes	Yes	Yes	No	No	No	Yes
[30]	Yes	Yes	No	Yes	Yes	No	Yes
[31]	No	Yes	Yes	No	No	No	Yes
[32–34, 34–36]	Yes	Yes	No	No	No	No	Yes
Our Survey	Yes	Yes	Yes	Yes	Yes	Yes	Yes

- Introduce over 115 representative works from 2010 - 2020. Provide detailed explanations of each category's most famous works and discuss inter-connections of all works in each category.

- Discuss the most challenging topic, Cross-Modality TL, which has never been discussed in any previous surveys.

- Present deep insights to current challenges and frontier of TL applications.

- This survey can be used as a guideline for professionals to develop TL models.

Finally, the remainder of this chapter is structured as follows: In Section-2.2, the author introduces a number of recent surveys on TL, and demonstrate the improvements made by our survey. And then, in Section-2.3, the author gives an overview of the survey, and this overview can help professional to find well-suited methods for different situations quickly. Secondly, in Section-2.4, the author first reviews the most recent TL works. In-between, the author also introduces some successful applications in industries. And then, the author presents the future trends and the open challenges in Section-2.5 and Section-2.6. Finally, the author concludes the article in Section-2.7.

2.2 Related Work

In this section, as shown in Table-2.1, the author reviews several surveys on TL in the past decade. Moreover, the author demonstrates the main differences in our survey to distinguish it from other recently published works.

Recently, some surveys of TL with informative contents are provided for readers from both the academies and the industries. These surveys [19, 23, 28–31, 33–38] categorize and review a wide range of TL techniques from different perspectives, such as algorithm types, applications, and the mixture of both.

First of all, the author introduces some widely known surveys for TL algorithms. The survey [23] gives readers a brief overview and detailed explanations of representative TL algorithms from 2000 to 2010. However, this work does not cover several newly introduced TF disciplines, such as TL with artificial neural networks, heterogeneous TL, and TL with adversarial networks. In addition, the survey [28] gives attention to more recent TL topics that are not discussed in [23]. It introduces and summarizes a number of homogeneous and heterogeneous transfer learning algorithms from 2010 - 2015.

More recently, another survey [29] gives special attention to homogeneous TL and reviews of state-of-the-art homogeneous TL algorithms and applications. It reviews homogeneous TL from two perspectives: the data and the model. However, some advanced topics are not covered in this survey, including but not limited to heterogeneous TL, reinforcement TL, and lifetime TL. Moreover, heterogeneous TL is specially discussed by the survey [30]. Recently, deep learning has received increasing attention from the TL community. A recent survey [31] focuses on TL with deep learning. It provides a formal definition of deep transfer learning and reviews current works in four deep TL disciplines: instance-based, mapping-based, network-based, and adversarial-based. Furthermore, there are some surveys [32–34, 34–36] particularly concentrate on TL applications in different fields: health care systems, sentiment analysis, remote sensing, recommendation systems, and signal processing.

Our work covers the most recent topics in the past decade, such as TL with deep learning, TL with artificial neural networks, TL with statistical methods, TL with lifelong learning, and TL applications. Moreover, our survey also discusses the most challenging topic, cross-modality, and distant domain TL, which are not well-investigated in other surveys. Furthermore, detailed explanations of each type TF discipline's representative methods are provided for readers to have a better understanding. What is more, TL-related applications and current trends of TL are also discussed.

Table 2.2: Terminology Definition

	Domains	Tasks	Modalities
Inductive TL	Same	Same	Same
Transductive TL	Same	Different but related	Same
Unsupervised TL	Different but related	Different but related	Same
Cross-Modality TL	Different	Different	Different

Table 2.3: Transfer Learning

Non-Cross-Modality	
Transductive Learning 2.4.1	
Feature-Based	[2, 3, 11, 17, 25, 27, 39–48]
Instance-Based	[1, 49–51]
Inductive Learning 2.4.2	
Feature-Based	[25, 46, 52–58]
Instance-Based	[59–65]
Parameter-Based	[5, 66–69]
Relation-Based	[1, 49–51]
Unsupervised Learning 2.4.4	
Feature-Based	[23, 70–72]
Negative Learning 2.4.5	
Problem Definition	[23, 28, 73–76]
Algorithms	[70, 77–79]
Cross-Modality	
Cross-Modality Learning 2.4.3	
Supervised Target Data	[9, 80]
Semi-supervised Target Data	[81–84]

2.3 Overview

In this section, the author gives an overview of all methods that are discussed in the survey. As shown in Table-2.3, the table can be used as an index to help professionals to quickly find the works related to their specific interests. Moreover, it is also helpful for selecting appropriate methods to solve given TL problems.

There are three steps to find the most suited methods for a given TL problem. Firstly, it is essential to decide if the given problem is a regular TL task or a cross-modality. For example, from text to image is a cross-modality task, and from image to image is a conventional TL task.

For regular TL problems, there are four categories. The first three categories can be defined by the source domain's label availability and the target domain. Moreover, negative learning can be defined by measuring the statistical distance between the source domain and the target feature domain. For cross-modality TL problems, there are two categories defined by the label availability in the target domain.

2.4 State of the Art

This section presents the state-of-the-art of TL in the past decade.

2.4.1 Transductive TL

Table 2.4: Transductive Learning

Transductive Learning	
Feature-Based	[2, 3, 11, 17, 25, 27, 39–48]
Instance-Based	[1, 49–51]

The definition of transductive learning [23] is: the tasks in the source and target domains are the same, but the domains may be different. Under this setting, the labeled data is only available in the source domain. Furthermore, there are two learning types in transductive transfer learning: learning on instances and learning on features. Moreover, the most widely known example of transductive learning is domain adaptation. In transductive TL, instance-based methods are not as popular as feature-based methods due to the limitations of its learning mechanism that is detailed in the following section. Therefore, the current mainstream of transductive TL is feature-based methods.

2.4.1.1 Learning on Instances

Primarily, algorithms of learning on instances are defined as transferring the knowledge in the source domain to the target domain by re-weighting or re-sampling source instances. Moreover, instance-based methods are built upon two strict assumptions:

1) the particular amount of training instances in the source domain are related to the target domain so that they can be reused, and 2) the conditional distributions of the source and the target domain are identical.

Importantly, not all the source data can be re-used for training the target model. Therefore, it is important to properly select samples that can benefit the task in the target domain. Firstly, [51] proposed a boosting method that leverages the concept of AdaBoost. Similarly, [1] proposed two novel approaches for instance re-weighting and instance selection based on the concept of PU learning and the in-target-domain probability. As shown in Figure-2.2, it first samples a small set $\hat{P}$ from unlabeled data P in the target domain as spies and labels all the instances $x \in P - \hat{P}$;n. as true. Then it labels $\hat{P} \cup U$ as false. A Naive Bayes (NB) classifier is then applied to $\hat{P}$ and U to identify a reliable negative set N_r based on the threshold b. The next step is to find the in-target-domain probability of $U_r = U - N_r$ by applying an Expectation Maximization (EM) algorithm. In Instance Selection (PUIS), the instances with higher in-target-domain probability are selected. Differently, Instance Weighting (PUIW) first calibrates the in target-domain-probability, and then use it as the sampling weights for training NB model.

However, methods similar to [1] are not efficient and heavily dependent on the preset values of the calibration parameters when the tasks have high-dimensional distributions. Moreover, some other instance-based adaptation models [49, 50] can handle tasks with have high-dimensional distributions. The core concept of this type of models is to adapt data in the source domain to the target domain by applying a logistic approximation.

More recently, [85] developed an instance-based multi-source transfer learning method

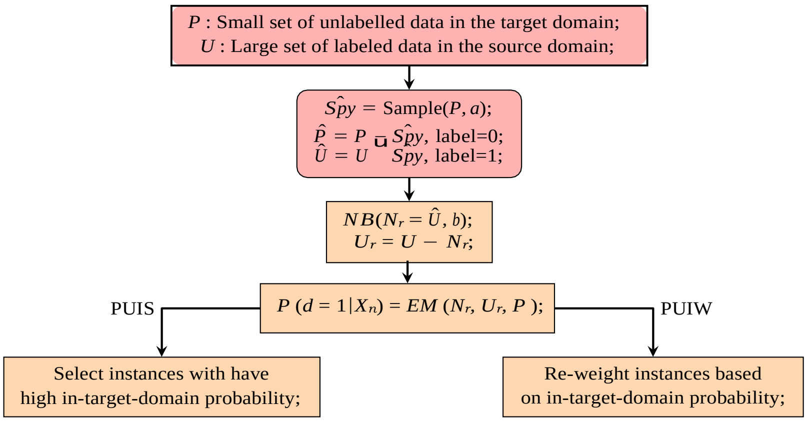

P : Small set of unlabelled data in the target domain;
U : Large set of labeled data in the source domain;
Spy = Sample(P, a);
P̂ = P ∪ Spy, label=0;
Û = U Spy, label=1;
NB(N_r = Û, b);
U_r = U − N_r;
P (d = 1|X_n) = EM (N_r, U_r, P);
PUIS
PUIW
Select instances with have high in-target-domain probability;
Re-weight instances based on in-target-domain probability;

Figure 2.2: PUIS & PUIW [1]

based on the maximal correlation analysis [86]. Notably, it does not require the data from source domains to train a target domain model. Instead, it only requires the pre-trained source domain models to construct a set of distributed networks as a feature extractor for the target domain data. By doing this, the computation of the training is significantly reduced. What is more, a novel maximal correlation metric [87] was introduced to measure the distribution distance. More than that, as shown below, it also proposed four rules for designing algorithm-specific TL algorithms. The four rules are:

- Minimize the weighted empirical loss over source and target domains.
- Assign balanced weights to data points, as focusing too much on specific data points leads to over-fitting caused by perturbations in the training data .
- Assign more weight to the target sample, since target data will be used for testing.
- Assign weights such that the performance gap between the domains is small.

Moreover, it also proposed a novel algorithm called GapBoost, which adjusts the instance weight matrix by applying on a novel domain distance measurement, $Y - Discrepancy$:

$$dist_Y(D_S, D_T) = sup|L_{D_S}(h) - L_{D_T}(h)|, h \in H,$$

where h is the optimal chosen learning model during each iteration in the training stage.

2.4.1.2 Learning on Features

However, those required conditions of instanced-based algorithms do not always hold in many real-world problems [17, 43, 57]. Alternatively, feature-based methods have been developed to solve the issues. Firstly, [25] introduced the idea of transferable features for deep neural networks. In general, learning on features only needs a weaker hypothesis: the distributions of the target domain and the source domain are similar. Intuitively, it tends to minimize the distribution mismatch between the source domain and target domain by transferring or re-representing features to another space. Generally, there are two types of feature-based transductive learning methods: data-centered methods [2, 11, 17, 39, 40] and subspace-centered methods [3, 42–45].

Generally, data-centered methods are to discover a uniform transformation that can convert the data from the source domain and the target domain to a domain-invariant space so that the distribution mismatch can be minimized without losing original information. However, so it does not work well when the target domain and the source domain have a large discrepancy. Differently, subspace-centered methods try to reduce the domain shift by manipulating sub-spaces of the source domain and the target domain. To do this, it is important to find the appropriate projections for the data in both domains.

Firstly, the idea of adaptation layer was proposed by [39]. It introduced a modified feedforward neural network, Domain Adaptive Neural Network (DaNN), with one adaptation layer. Importantly, the loss function is contains two parts: the general loss

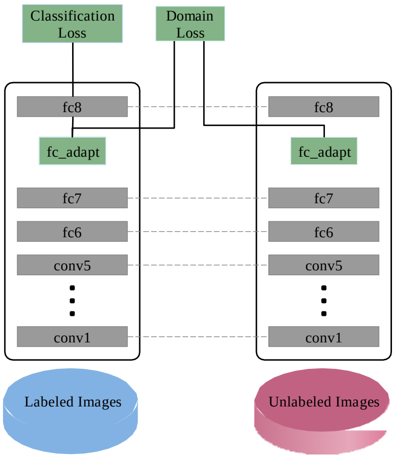


Figure 2.3: Deep Domain Confusion [2], it is an AlexNet-based architecture with one adaptation layer and an additional domain confusion loss (MMD-based) was proposed to learn a semantically meaningful and domain invariant representation.

and the MMD loss. Additionally, the MMD loss is used to evaluate the distribution mismatch between the source domain and the target domain. The model has produced better performance than similar models [26, 88]. However, it is a very shallow and simple model, so the performance is limited. Furthermore, several studies have approved the deep neural networks can learn much more transferable features, so we would like to benefit from the deeper features. To explore the potential of DaNN, a number of novel methods were proposed [2, 3, 11, 40, 41]. As illustrated in Figure-2.3, Deep Domain Confusion (DDC) [2], an AlexNet-based [10] Convolutional Neural Network (CNN) with one adaptation layer and an additional domain confusion loss (MMD-based) was proposed to learn a semantically meaningful and domain invariant representation. Additionally, the evaluation metric can also be used to determine the position and the dimensionality of the adaptation layer. Furthermore, [42, 43] improved the performance of [2] by introducing weighted-MMD with weight regularizer.

Moreover, [17] added another term, CORAL loss, to the regular loss function to produce even better results. In this method, CORAL loss, f_{CORAL}, is defined as the distance between the second-order covariances of the source and the target features:

$$f_{CORAL} = \frac{1}{4d^2} \| C_S - C_T \|_F^2$$

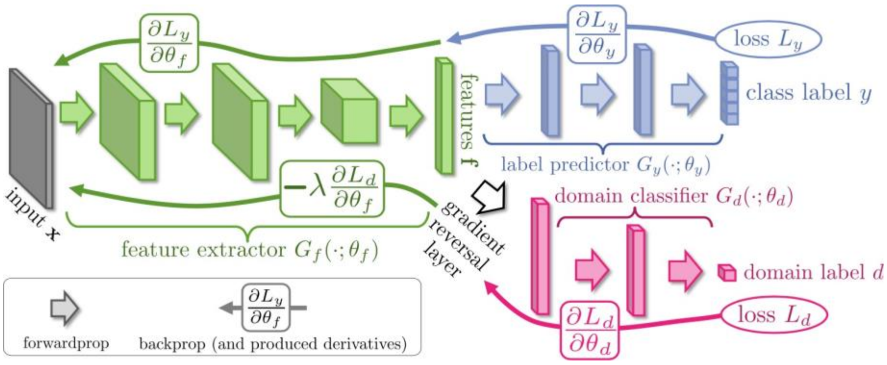

$\frac{\partial L_y}{\partial \theta_f}$
input x
features $\mathbf{f}$
$-\lambda \frac{\partial L_d}{\partial \theta_f}$
feature extractor $G_f(\cdot;\theta_f)$
$\frac{\partial L_y}{\partial \theta_y}$
loss L_y
class label y
label predictor $G_y(\cdot;\theta_y)$
gradient reversal layer
domain classifier $G_d(\cdot;\theta_d)$
domain label d
loss L_d
$\frac{\partial L_d}{\partial \theta_d}$
forwardprop
$\frac{\partial L_y}{\partial \theta_f}$
backprop (and produced derivatives)

Figure 2.4: Gradient Reversal [3], it has three components: a feature extractor (green), a label predictor (blue), and a domain classifier (blue).

However, the buried features in the deep layers could be highly task-specific, so that they cannot be safely transferred to new tasks. To solve this issue, another framework was proposed by [11]. It introduced a novel framework, deep adaptation networks (DAN), to enhance the feature transferability and reduce the domain shift. Differently, multi-kernel MMD is used to close the distribution mismatch between the source domain and the target domain, and multiple adaptation layers are applied to improve the performance. As a classic example of multi-kernel MMD-based architectures, The Domain Adaptive Hash (DAH) network [45] combines hashing techniques and multi-kernel MMD. To the best of our knowledge, it is the first research that exploits the feature learning capabilities of neural networks to learn representative hash codes to address the domain adaptation problem. Particularly, hashing techniques can also convert the high dimensional data into binary codes, so it will be easier to access and store. In addition, there are more models [27, 46, 47] that have used adaptation layer. Especially, [48] is able to transfer across domains and tasks simultaneously.

Differently, [3] wishes to learn the underlying features that combine the discriminativeness and domain-invariance. The network architecture is shown in Figure-2.4. It has one feature extractor and two sub-classifiers. The underlying features can be learned by training two classifiers simultaneously, label predictor and domain classifier. The feature extractor can minimize the loss of the label predictor and maximize the loss of the domain classifier to make sure the features are domain-invariant. The loss function is constructed as:

$$E(\theta_f, \theta_y, \theta_d) = \sum_{i=1, d_i=0}^{N} L_y^i(\theta_f, \theta_y) - \lambda \sum_{i=1}^{N} L_d^i(\theta_f, \theta_d),$$

where L_y is the loss for label prediction, L_d is the loss for domain classification. However, the standard stochastic gradient descent does not fit this procedure because of the negative sign in front of the L_d loss. To solve this problem, gradient reversal layer (GRL):

$$R_\lambda(x) = x,$$

$$\frac{dR_\lambda}{dx} = -\lambda \mathbf{I},$$

was introduced to smoothly connect the feature extractor and domain classifier. Next, the GRL function is plugged into the loss function:

$$E(\theta_f, \theta_y, \theta_d) = \sum_{i=1, d_i=0}^{N} L_y^i(\theta_f, \theta_y) + \sum_{i=1}^{N} L_d^i(G_d(R_\lambda(G_f(x_i; \theta_f)); \theta_d), y_i) \tag{2.1}$$

2.4.2 Inductive TL

Table 2.5: Inductive Learning

Inductive Learning	
Feature-Based	[25, 46, 52–57]
Instance-Based	[59–65]
Parameter-Based	[5, 66–69]
Relation-Based	[1, 49–51]

Unlike transductive learning, inductive learning is defined as: the tasks in the source domain and the target domain are different regardless if the domains are the same or not. Under this setting, the well-labeled data is usually available in the target domain, no matter the well-labeled data is available or unavailable in the source domain. Particularly, the main focus is on the former. In this case, inductive learning is similar to multi-task learning, but it only concentrates on the target task. Differently, when there is no labeled data in the source domain, inductive learning is close to self-taught learning. The information is hidden in the source domain, so it cannot be used directly. Commonly, inductive TL aims to develop a target model with a small set of well-labeled data in the target domain.

Additionally, there are four learning types in inductive learning: learning on instances, learning on features, learning on parameters, and learning on relations. Furthermore, the first three types of methods are the mainstream in inductive learning, while relation-based methods are not very common.

2.4.2.1 Learning on Instances

Generally, the training data in the source domain are more or less out-dated, and processing new data is very costly. Inductive TL aims to train an accurate model with only a tiny amount of well-labeled training data in the target domain. Moreover,

Algorithm 1: TrAdaBoost

```
Input: Two labeled training sets $T_d$ and $T_s$;
       The unlabeled testing set $S$;
Initialize: Learner, $F$;
            The number of iterations, $N$;
            Weight Vector, $W^1$;
for $t = 1, ...., N$ do
    1. Set $P^t = w^t / \sum_{i=1}^{N} w_i^t$.
    2. Apply Learner, $F_t(X) = Y$.
    3. Calculate the error:
       $E_t = \sum_{i=n+1}^{n+m} \frac{w_i^t |h_t(x_i) - c(x_i)|}{\sum_{n+1}^{n+m} w_i^t}$.
    4. Set $\beta_t = E_t/(1 - E_t)$ and $\beta = 1/(1 + \sqrt{2 \ln n/N})$.
    5. Update the new weight vector:
       $w_i^{t+1} = \begin{cases} w_i^t \beta^{|h_t(x_i) - c(x_i)|} \\ w_i^t \beta_t^{-|h_t(x_i) - c(x_i)|}. \end{cases}$
end
Output: $F_t(x) = \begin{cases} 1, & \prod_{t=\lceil N/2 \rceil}^{N} \beta_t^{-h_t(x)} \geq \prod_{t=\lceil N/2 \rceil}^{N} \beta_t^{-\frac{1}{2}} \\ 0, & otherwise \end{cases}$
```

the key of this type of methods is finding which part of the old data can be adapted to train a new model in the target domain. One of the most famous instance-based methods in inductive learning is TrAdaBoost [59], an AdaBoost [60]-based transfer learning algorithm. Conceptually, it extracts useful information in the source domain by iteratively re-weighting the source domain instances. Firstly, it employs a few labeled new data, called same-distribution data T_s, to evaluate the value of each old the old data in the source domain. Furthermore, the instances with low value are classified as diff-distribution data T_d. And then, it combines T_d, T_s, and unlabeled data S to train a new model for the target task. However, the re-weight procedure of TrAdaBoost is not the same as AdaBoost. Additionally, it increases the weights of incorrectly predicted instances in T_d, while decreases the weights of correctly predicted instances in T_s. Similarly, [87] proposed GapBoost, a novel multi-source boost method for transfer learning.

Recently, several algorithms inspired by TrAdaBoost have pushed the performance to a new level. Firstly, one of the shortcomings of TrAdaBoost is only using one type of base learner to train the model in the target domain, but there might be other base learners that can give better performance. To address this issue, [61, 62] choose to employ different base learners to improve the performance on specific tasks. Secondly, the original TrAdaBoost algorithm only uses one source domain for the knowledge transfer. However, the knowledge is not always enough from a single source domain. In order to overcome this shortcoming, [63, 89, 90] take advantage of combining multiple source data sets to avoid negative learning. Additionally, [90] can decide which sources are helpful to build the model in the target domain by iteratively performing two types of boosting: 1) individual boosting for instances and 2) task-based boosting. It increases the weights of incorrectly predicted instances, and it also performs a task-based boosting that can enhance the instances from the tasks that have higher transferability. Unlike TrAdaBoost, it keeps all the base learners can improve the performance of the model because the early iterations fit the majority of the data while the later iterations focus on more in-depth details. Furthermore, there are also researches [64, 65] that improve the model with dynamic weight update methods.

Overall, re-weighting instances iteratively is a proven way to enhance inductive learning models' performance, yet some other researchers hold different opinions. Commonly, certain parts of the differently distributed data T_d could help training the model in the target domain, yet certain parts could also be harmful. Moreover, there are no simple methods to measure the transferability of the source data sets accurately. Therefore, some algorithms [91, 92] intend to remove all the different distribution data instead of assigning small weights to them.

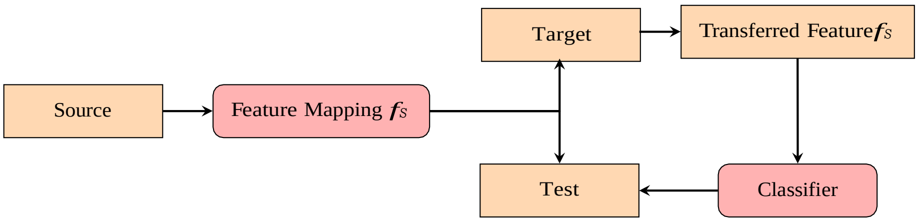


Figure 2.5: Feature-Based Inductive Learning [4], it requires a large set of labeled data from one or multiple source domains, and a small amount labeled data from the target domain. The core idea is to train three separate model by augmenting the original data into three sets, namely, source-specific, target-specific, and general-specific.

2.4.2.2 Learning on Features

Commonly, feature-based inductive transfer learning algorithms [25, 46, 52–57] wish to extract shared features to minimize domain divergence and model error. According to the types of source data sets, feature-based algorithms can be classified into two categories: supervised and unsupervised. Firstly, supervised algorithms [25, 46, 52–56] are similar to multi-task learning, which combines a sufficient amount of labeled source data and a tiny amount of labeled target data to train a high-quality model in the target domain. However, multi-task learning tends to learn all the tasks simultaneously, while inductive transfer learning only focuses on the target task. Differently, unsupervised algorithms [57, 87] are more powerful but difficult to train.

Primarily, most feature-based inductive transfer learning methods focus on finding domain-invariant features. In other words, the problems can be converted into how to effectively extract features that can reduce the divergence between the source domain and the target domain [23]. [52] introduces a simple, fully supervised approach with feature-augmentation. Firstly, it requires a large set of labeled data from one or multiple source domains, and a small amount labeled data from the target domain. And then, it trains three separate model by augmenting the original data into three sets, namely, source-specific, target-specific, and general-specific. Additionally, three sets of weights of those data sets are denoted as W_s, W_t, W_g. Moreover, W_s represents the sum of the ”source” and ”general” features, W_t represents the sum of the ”target” and ”general” features. And the feature-augmented weights are regularized by $|W_g|^2 + |W_s - W_g|^2 + |W_t - W_g|^2$. Finally, minimizing the sum of the equation will find the features that can minimize the divergence. Moreover, as shown in Figure-2.5, [4] proposed a framework to justify the effectiveness of feature-based inductive transfer learning. Firstly, it constructs a feature mapping, F , for the source domain data. Then use this mapping to transfer the target domain data to the same feature space. After that, it trains a discriminative classification model based on the feature extracted by F . Besides, the mapping learned in the first step will also be used to convert the test data into the same feature space as the training data.

Recently, several works [25, 46, 56, 57] have evaluated the combination of GANs and transfer learning. Initially, this kind of methods aim to free human from hand-designing networks for extracting shared features. For example, [56] aims to find features that are 1) discriminative for the main learning task in the source domain and 2) domain-invariant by implementing the idea of GANs. Moreover, these features are considered ideal for cross-domain transfer when models cannot identify the original domain of the inputs [25]. As shown in Figure-2.6, the model includes three main components, domain classifier G_f, predictor P , and feature extractor G_f. The final goal is to learn the mapping (M) to predict unknown instances in the target domain with low risk. Furthermore, the risk is defined as follow:

$$R_{D_T}(M) = Pr_{(x,y)\ D_T\ (M(x)/=y)}$$

where D_T represents the target domain, and M represents the mapping from the features to the labels.

Similar to the typical GANs model, domain classifier and predictor will be adversarial to each other. As shown in Figure 2.6, the parameters of the domain classifier are trained to minimize the loss during the training. The feature extractor parameters are optimized to minimize the loss of the predictor f_c and maximize the loss of the domain classifier f_d. Therefore, the loss of the model is constructed by two terms:

$$f = f_c(D_s, y_s) + \lambda f_d(D_s, D_t)$$

where D_s represents the source domain, D_t represents the target domain, and λ is the learning coefficient.

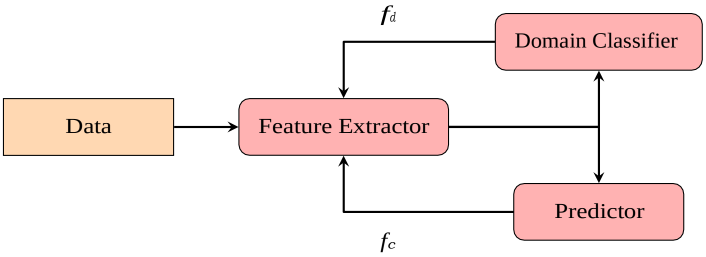


Figure 2.6: GANs-Based TL

2.4.2.3 Learning on Parameters

Generally, parameter-based approaches [5, 66–69] are based on the assumption that there are shared-parameters in models from source domains and the target domain. Thus, this type of methods are not suitable for the cases with a significant domain shift. Under this setting, parameter-based methods can be easily derived from multi-task learning methods. However, multi-task learning is usually focused on learning all the tasks simultaneously, while parameter-based transfer learning is only focused on optimizing the target task. Thus, the loss functions for all the tasks are the same in multi-task learning, but the loss function in the target domain has greater weights in the transfer learning.

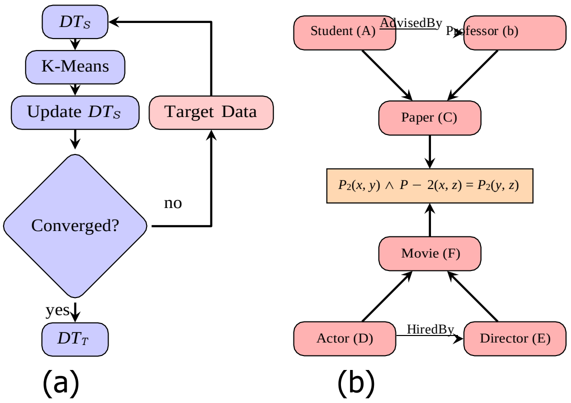


Figure 2.7: 2.7a: TransEMDT [5], it first trains a decision tree model based on the source data (DT_S). Secondly, it feeds a small amount of the labeled target data into DT_S, and the prediction is used as initial clusters for K-Means model. 2.7b: Markov Logic Network [6–8], the Markov Logic Network can be demonstrated by finding similar relationships from two different domains to construct a mapping from the source domain to the target domain.

Firstly, [5] introduced a decision tree embedded transfer learning framework. TransEMDT (Transfer learning EMbedded Decision Tree) aims to address supervised transfer learning problems. As shown in Figure-2.7a, it first trains a decision tree with the source data (DT_S). Secondly, it feeds a small amount of the labeled target data into DT_S , and the prediction is used as initial clusters for K-Means model. After that, the parameters of DT_S is updated. Then the previous steps will be repeated until it converges. Finally, the output will be the decision tree for DT_T . Similar to TransEMDT, [67] proposed another framework, TransRKELM (Transfer learning Reduced Kernel Extreme Learning Machine), which uses RKELM to build an initial activity recognition model. Furthermore, several algorithms [68] have achieved promising performance by modifying SVM (Support Vector Machine). Typically, they assume that weight vectors of SVM contains two components: $W = W_S + W_D$, where W_S represents weight vectors that are shared across the source and the target domains, while W_D represents domain-specific weight vectors. In general, the traditional discriminative query strategy results in poor performance when there is a significant distribution mismatch between the source domain and the target domain. Some studies [68, 69] applied the generative query strategy to overcome this shortcoming. Moreover, [69] extended binary learning method to multiclass problems by implementing the one-vs-all approach. Furthermore, [66] presented Multilinear Relationship Networks(MRN). It can prevent negative transfer in the feature layers by jointly learning transferable parameters and multilinear relationships.

2.4.2.4 Learning on Relations

Comparing to other topics in inductive TL, relation-based transfer learning is not very popular. Unlike the other three types of learning methods, relation-based transfer learning methods do not assume the source data and the target data to be independent and identically distributed (i.i.d). This makes relation-based methods much more flexible and robust than traditional methods. However, there are not many studies on this topic in recent years. Moreover, most of this type of algorithms are built based on statistical learning techniques. The idea behind relation-based transfer learning is that similar relations exist in different domains. For example, the data in the source domain contains images of a professor giving a lecture to students, and the data in the target domain contains images of a manager giving a speech to employees. Although two sets of images describe different objects, they have the same relation.

Some studies [6–8] have proposed to use Markov Logic Networks. As shown in Figure-2.7b, the Markov Logic Network can be demonstrated by finding similar relationships from two different domains to construct a mapping from the source domain to the target domain.

2.4.3 Cross-Modality Transfer Learning

Table 2.6: Cross-Modality Transfer Learning

Cross-Modality Learning	
Supervised Target Data	[9, 80]
Semi-supervised Target Data	[81–83]

Commonly, most TL algorithms require more or less the connection in feature spaces or label spaces between the source and the target domain. In other words, knowledge transfer can only be performed when the source data and the target data are in the same modality, such as image, audio, and text. Unlike all other TL methods, Cross-Modality Transfer Learning (CMTL) is one of TL's most challenging topics. It assumes that the source and the target domain's feature spaces are entirely different, such as from text to image, from audio to text, and from image to audio. Moreover, the label spaces between the source and the target domain can also be different.

Intuitively, CMTL is inspired by humans' ability to generalize knowledge from one subject to another by building a bridge with other subjects. For example, a child who has read an article with descriptions of monkeys, and he has never seen any monkeys or images of monkeys. However, it is very possible that the child can recognize a monkey based on that article's knowledge. In this case, a child can transfer the knowledge from text data to image data using knowledge in other different domains. Theoretically, two seemingly unrelated domains can be connected by one or multiple bridge domains with overlapping semantic information. However, this type of learning behavior is difficult for machines to mimic due to the challenge in selecting appropriate intermediate domains as the bridge. Moreover, there are two types of CMTL algorithms: CMTL with Supervised Target Data and CMTL with Supervised Target Data.

2.4.3.1 CMTL with Supervised Target Data

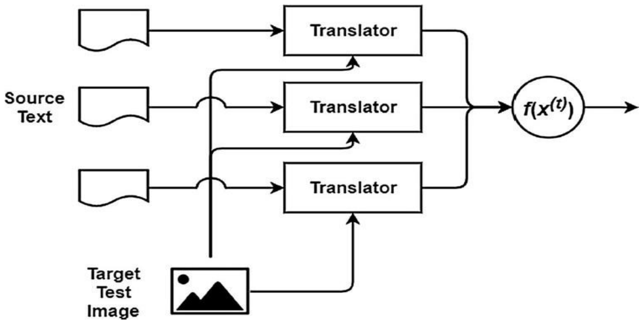


Figure 2.8: Text-to-Image [9], CMTL transfers between knowledge between text files and images with multiple translators.

This section discusses several text-to-image (TTI) DDTL methods, which require a small set of labeled image target data. Importantly, image classification tasks currently have two challenges: 1) labeled image data is relatively scarce and expensive to collect, and 2) features of image data lack semantic meaning for class prediction as they represent visual features rather than conceptual ones. Moreover, labeled text data is often more accessible than labeled image data, and text features have more semantic meaning for predicting a class label.

Firstly, Translated Learning via Risk Minimization (TLRisk) was introduced by [80]. It proposed an asymmetric architecture to map the features in the source domain to the target domain. Moreover, it uses a language model [93] and the nearest neighbor method to connect the text source data and the image target data. Moreover, for a smooth feature transition, it builds a translator by applying the Markov chain. The source features and the target features are modeled by two different Markov chains, which can be bridged with intermediate data. In other words, the translation is done by learning a probabilistic model that uses cooccurrence data as a bridge between the source and target feature spaces. Finally, it uses a variant of the risk minimization model to produce the final label prediction. This method outputs promising results that are better than the baseline model trained on only target data. However, the computational cost of TLRisk is very expensive due to the risk function estimation and dynamic programming.

To decrease the computational cost, [9] proposed another method for text-to-image (TTI) classification. In this study, the source domain is text data, and the target domain is image data. This method implements a novel transition method, translator, to build a bridge from text to images. It requires labeled source text data, text-image cooccurrence data, and a small amount of labeled image target data. This method uses TL to exploit such text data to improve image classification. Therefore, this problem is converted to how to relate the text to semantic knowledge transfer images. Moreover, this method uses a text-image cooccurrence matrix that contains images and the text that occurs with them on the same webpage. Cooccurrence information is effective because of the assumption that the text around an image describes the concepts in such an image. This cooccurrence information is relatively inexpensive to collect and serves as a bridge to learn the correspondence for translating the semantic information between the source text and the target image. This translation is achieved by the form of a feature transformation called a ”semantic translator function.” This translator takes the source, the target, and the cooccurrence data and learns the correspondence between the source text and the target images through the cooccurrence bridge. Each translator for the source text contains a ”topic space,” a common subspace associated with the translation data. As shown in Figure-2.8, there are a number of translators combined to form the final decision function $f(x^{(t)})$. Furthermore, this method bypasses the performances conducted by [80, 81] and other benchmark models trained with only target data, and it yields state-of-art accuracy with only a little target training data.

2.4.3.2 CMTL with Semi-supervised Target Data

Unlike CMTL with supervised target data, several methods can take labeled and unlabeled target data to improve the classification performance.

Firstly, [81]proposed a heterogeneous TL for Image Classification (HTLIC) method that can take in semi-supervised source data and target data. Moreover, it aims to enhance a target image classification task with limited labeled data by exploiting semantic knowledge derived from unlabeled text documents and unlabeled annotated images from an auxiliary source. The unlabeled auxiliary data is relatively inexpensive to collect and it can enhance target image classification performance. It aims to find the relationship between unlabeled source text data and the semi-supervised image target data using auxiliary data with related semantic information. Furthermore, the connection is discovered using a two-layer bipartite graph where the top layer represents the relationship between the images and the tags, while the bottom layer represents the relationship between the tags and the documents. The feature space gap between the source domain and the target domain can be reduced. Moreover, more shared semantic information can be discovered with this bridge in low-level features with semantic analysis [94]. Unlike previous methods, HTLIC does not use a Markov chain to achieve the classification task. Instead, it applies traditional support vector machines (SVMs) [95] to make the final predictions. As the main improvement of this method, it proposed an efficient way to utilize semi-supervised target data to produce promising classification accuracy.

Furthermore, [4] first introduced the idea of using co-occurrence information between two different domains. And then, [96] proposed Co-occurrence Transfer Learning (CT-Learn) for knowledge transfer between text data and image data. More importantly, it enables the knowledge transfer from multiple domains, significantly improving the target classification accuracy with appropriate source domain selection. Unlike the previous methods [81], CT-Learn first uses the co-occurrence information between the text data and image data to create a joint transition probability matrix P:

$$P = \begin{bmatrix} \lambda_{1,1}P^{(1,1)} & \lambda_{1,2}P^{(1,2)} & \dots & \lambda_{1,N}P^{(1,N)} \\ \lambda_{2,1}P^{(1,1)} & \lambda_{2,2}P^{(2,2)} & \dots & \lambda_{2,N}P^{(2,N)} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{N,1}P^{(N,1)} & \lambda_{N,2}P^{(N,2)} & \dots & \lambda_{N,N}P^{(N,N)} \end{bmatrix}. \tag{2.2}$$

This matrix is constructed using intra-relationships and inter-relationships for all the co-occurrence, labeled, and unlabeled instances across both domains. Moreover, the intra-relationships are calculated by the affinity of the intrinsic manifold structure between the *ith* domain, and the inter-relationships are calculated by using the co-occurrence information. The diagonal elements represent intra-relationships, and other elements indicate inter-relationships between the *ith* and the *jth* domains. The weights λ decide the amount of transferable knowledge between domains, which shares a similar idea of learning rate in artificial neural networks. Furthermore, after extract the inter-relationships and the intra-relationships, it creates a coupled Markov chain based on a random walk with a restart. Different from TLRisk [80], CT-Learn applies a variant of regular Morkov chain to adapt multiple source domains. Moreover, most previous methods can only handle binary classification problems, but CT-Learn can deal with binary and multi-class classification problems. Finally, CT-Learn performed the highest accuracy on most benchmark data sets.

Table 2.7: Unsupervised Transfer Learning

Unsupervised Learning	
Feature-Based	[23, 70–72]

2.4.4 Unsupervised Transfer Learning

Primarily, the idea of transfer learning was proposed to solve the issue of lacking data. Moreover, many transfer learning methods have successfully generalized machine learning techniques to practical and performance-critical problems. However, most algorithms are focused on supervised cases and semi-unsupervised cases. In general, supervised algorithms cannot deal with cases where we do not even have enough labeled in the source domains.

Conceptually, unsupervised TL is defined as no labeled data in both the source domain and the target domain. This type of methods are beneficial to tasks that are unique and special, so a sufficient amount of labeled data from both the source domain and the target domain are not accessible. However, researchers have not favored this topic due to some barriers that make it difficult to apply to real-world tasks. Generally, there is only one sub-field under this setting: feature-based learning. Additionally, unsupervised transfer learning is also termed as self-taught learning by many researchers and scholars.

2.4.4.1 Learning on Features

Firstly, a few methods [70, 71] for clustering and dimensional reduction problems were summarized by [23]. The concept of Self-taught Clustering (STC) was introduced by [71], which aims to perform clustering on a small set of unlabeled target data with the help of a sufficient amount unlabeled in the source domains. In theory, STC tends to convert data sets in different domains into a common feature space, which can utilize the source data to cluster the target data. Moreover, proposed Transferred Discriminative Analysis (TDA) was proposed by [70]. It can generate pseudo-class labels for the target data by applying clustering methods.

Furthermore, a novel self-taught learning algorithm was introduced by [72]. It uses sparse coding to construct higher-level features using the unlabeled data. Moreover, this algorithm had been shown to improve the performance of classification tasks significantly.

2.4.5 Negative Transfer Learning

Table 2.8: Negative Transfer Learning

Negative Learning	
Problem Definition	[23, 28, 73–76]
Algorithms	[70, 77–79]

Certainly, transfer learning has successfully solved the issue of lacking training data in many real-world applications. However, it also has one shortcoming: negative transfer. Commonly, negative transfer occurs when transferring too much unrelated knowledge from the source domains. Despite its pervasiveness, negative transfer is usually described in an informal manner, lacking rigorous definition, careful analysis, or systematic treatment. Firstly, there are numerous survey papers [23, 28, 73] have discussed this issue in many TL disciplines. Furthermore, some researches [74–76] have recognized it in many real-world applications. In this section, the author introduces some of the works that address negative learning.

First of all, typical TL assumes that the target domain and the source domain are different but related, so some common instances or features can be transferred between different domains. However, it limits TL from being applied to cases where the source and the target are very loosely connected. To address this issue, some works focus on transferring knowledge between two distant domains. Firstly, an instance-based algorithm [77], transitive transfer learning (TTL). It transfers knowledge between text data in the source domain and the image data in the target domain by using annotate image data as the knowledge bridge. However, this algorithm is very situational and case-dependent. Moreover, another feature-based method [78] was proposed to deal with scarce satellite image data. It predicts the poverty based on the daytime satellite image by transferring knowledge of an object classification task with the help of some nighttime light intensity information as a bridge. The main contribution of this method is to use similar data with different conditions to connect two different domains. Moreover, an instance-based distant domain transfer learning (DDTL) algorithm [79] uses several intermediate domains to bridge the source and the target. More specifically, it first uses an auto-encoder pair to select instances from the source domain and the intermediate domains, and it also learns high-level representations for data in different domains. After that, it trains a CNN model by using the selected instances and representations. Importantly, this method can be simply generalized to different tasks and produce fairly decent results. However, there are some challenges need to be addressed. Firstly, most chosen instances are from the intermediate domains and only a little from the source domain. Furthermore, it makes the source data seem unnecessary. The second, it assumes that there is a sufficient amount of intermediate domain data so we can find enough samples to build the bridge connecting the source and the target domains. In some cases, enough intermediate domains might not be accessible.

Furthermore, a study [97] first derived a novel definition of negative from three different perspectives, the chosen model, the divergence between the joint distributions, and the size of labeled target data, respectively. More importantly, it proposed a new term, negative transfer gap (NTG), to quantify the effect of negative transfer. It then introduced a novel GANs-based instance re-weighting algorithm to select useful samples from the source domain.

2.5 The Frontier of TL

Table 2.9: The Frontier of TL Applications

Transfer Learning Applications		
Signal Processing	Transductive TL [18, 98–103]	Distant Transfer [77, 79]
Sentiment Analysis	Inductive TL [104–107]	Transductive TL [108, 109]
Health System	Inductive TL[18, 19, 110–112]	Transuctive TL[113]
CPS	Inductive TL[114–116]	Transuctive TL[117]

In this section, the author presents the current trends in TL from two aspects: TL algorithms and TL applications. For TL algorithms, the author introduces several fields in TL that attract most attention. For TL applications, the author demonstrates various applications spanning multiple TL disciplines. Moreover, the main attention of the algorithm level is in solving the issues of insufficient data and distant domain transfer by conducting experiments that usually step ahead of making real productions.

Therefore, assumptions made in experiments do not always hold in real-world problems. Differently, real-world applications focus more on applying TL models with stable and promising performances, so methods with pre-assumptions cannot be used.

2.5.1 The frontier of Transductive TL

First of all, domain adaptation, a sub-field of transductive TL, is the most active area. It tends to solve problems where only have a sufficient amount of labeled source data and unlabeled target data for the training process. Therefore, domain adaptation methods can be categorized into a cluster of semi-supervised learning algorithms. Moreover, this semi-supervised manner gains more focuses than other TL topics do. Currently, existing domain adaptation algorithms aim to close the marginal distribution distance or conditional distribution distance in two ways: symmetrical training and asymmetrical training. The first, symmetrical training [2, 17] means that there are two models with identical structures for the source and the target domains. It is commonly applied to feature-based algorithms. The advantages of symmetrical training are: 1) easy to train, 2) fast convergence, and 3) robustness with small source data sets. However, it also suffers from a significant shortcoming: performance decrease due to large domain discrepancy. Moreover, asymmetrical training [77, 79] is related to the cases where the structures of the source model and Target model are not identical but have some common layers. In general, it is applied in instance-based domain adaptation algorithms. Furthermore, it can handle a large domain shift by selecting statistically similar instances from multiple source domains. However, with multiple source domains, asymmetrical training suffers from difficulties in the training and non-convergence.

Moreover, there are two common learning types of domain adaptation algorithms: feature-based and instance-based. Feature-based is the mainstream in the domain adaptation area since there is no labeled target data. Generally, feature-based methods aim to utilize all training samples from the source and the target by extracting shared features or closing the feature distribution distance. To extract common features, most algorithms first calculate the distance between low-level features from the source domain and the target domain with a distribution distance metric through each iteration. The next step is to select or re-weight the features based on the distribution distance to learn high-level feature combinations. Similarly, some feature-based algorithms tend to discover more shared features by converting features from different domains into a novel feature space where the distance of different features is small. Feature-based methods can carry out state-of-art performance when the source and the target have strong connections. However, the performance can drop if there is only a small amount of similar data samples across domains because a large number of different samples can overfit the model.

Differently, instance-based algorithms aim to select similar instances from different domains to ensure a safe and quality knowledge transfer. Combining instance re-weighting and distribution distance metric is the most commonly used technique in instance-based methods. Also, there are two different types of instance re-weighting: soft re-weighting and hard re-weighting. Firstly, soft re-weighting does not eliminate any instance. Instead, it just sets the weights of dissimilar instances to extremely small values. On the contrary, hard re-weighting eliminates all dissimilar samples by setting the weights to zero. With the selection procedure, training samples have a more reliable connection, and they can avoid the performance drop due to large domain discrepancy. Besides, instance-based methods can output relatively more stable performance. However, the performance can be disappointing when the volume of the source domain data is small because the number of selected instances can be insufficient if the domain distance is far.

2.5.2 The Frontier of Inductive TL

Generally, there are two main types of inductive TL learning algorithms: multi-source TL and self-taught TL. In common, both learning algorithms require labeled target data for the training process. Moreover, multi-source learning also needs labeled source data, while self-taught does not rely on labeled source data. Furthermore, multi-source learning attracts more attention due to its stable performance.

The main idea of multi-source learning is to take the advantage of multiple source domains. It is difficult to extract enough shared information from a single source domain in real-world problems due to the distribution discrepancy. Therefore, we aim to utilize multiple source domains to discover common features from each source domain and combine them to develop a source domain model. Moreover, this type of algorithms are usually stable and robust, but they are also computationally expensive due to the quantity of data from various domains. Under the setting of this type of algorithms, instance-based learning methods are more preferred than feature-based algorithms because the number of source training samples is sufficient for the training process. Furthermore, multi-source TL is closely related to supervised multi-task learning, another favored non-transfer ML technique. They both utilize multiple data sets from different domains and tasks. However, multi-task learning aims to improve the models in all different domains by sharing data sets. Differently, multi-source TL only focuses on the model in the target domain for the target task. Therefore, multi-task learning achieves better overall performances for multiple domains, and multi-source learning carries out better performance for a model in a specific domain.

Unlike multi-source TL, self-taught TL only requires labeled data from the target domain, which is more powerful but more costly and challenging to train. Moreover, feature-based learning methods and instance-based methods are both available in self-taught TL. In feature-based methods, unsupervised feature construction is required since there are no labels for the source domain data. The most commonly used unsupervised feature construction is sparse coding, which can be treated as a two-step minimization problem. In instance-based methods, the original TrAdaBoost [89] is the cornerstone of many advance self-taught TL algorithms, including but not limited to multi-source TrAdaBoost, weighted TrAdaBoost, and multi-class Boost. Furthermore, instance-based methods are generally easier to train because the convergence of unsupervised feature construction is not always guaranteed.

2.5.3 The Frontier of Distant Domain TL

Recently, insufficient training data and domain distribution mismatch have become the two most difficult challenges in ML. To address these two issues, TL has attracted more attention due to its training efficiency and domain shift robustness. However, transfer learning also suffers from a critical issue, negative transfer [75]. It significantly limits the use and performance of transfer learning. This section introduces some related works in three fields: conventional transfer learning, DDTL, and multi-task learning.

Firstly, TL aims to find and transfer the common knowledge in the source domain and the target domain. Furthermore, a research [46] expands the use of TL from traditional machine learning models to deep neural networks. Typically, there are two types of accessible TL: feature-based and instance-based. Moreover, both types focus on closing the distribution distance between the source domain and the target domain. In instance-based algorithms, the goal is to discover source instances similar to target instances to eliminate the highly unrelated source samples. Differently, feature-based algorithms aim to map source features and target features into a common feature space where the distribution mismatch is minimized. However, both of them naturally assume that the source domain and the target domain share a reasonably strong connection. Unlike conventional transfer learning, our work can transfer knowledge between different domains and tasks that are not closely related.

Secondly, most DDTL algorithms are similar to multi-task learning [118], which also benefits from shared knowledge in multiple different but related domains. Generally, multi-task learning tends to improve the performance on all the tasks. Differently, DDTL only focuses on using the knowledge in other domains to improve the target task's performance on the target domain.

Lastly, most previous studies of DDTL focus on instance-based methods and tend to take advantage of massive related source data. Firstly, [77] introduced an instance-based algorithm, transitive transfer learning (TTL). It transfers knowledge between text data in the source domain and the image data in the target domain using annotate image data as a bridge. However, this algorithm is highly case-dependent and unstable on performance. Similarly, another instance-based algorithm was introduced by [79]. It proposed a novel instance selection method, Selective Learning Algorithm

(SLA). Moreover, SLA can select helpful instances from many unrelated intermediate domains to expand the volume of the source data. However, this algorithm mainly aims to handle binary classification problems. Furthermore, a feature-based method[78] can deal with scarce satellite image data. It predicts the poverty based on the daytime satellite image by transferring knowledge learned from an object classification tasks with the help of some nighttime light intensity information as a bridge. However, this method heavily relies on a massive amount of labeled intermediate training data, which can be too expensive to apply. Unlike existing DDTL algorithms, a novel feature-based [84] method benefits from multiple unlabeled source domains data with significant discrepancies. Furthermore, it can also handle multi-class classification and consistently produce promising results.

2.5.4 The Frontier of TL Applications

In real-world problems, the most frequently and successfully applied ML technique is conventional supervised learning. After that, TL is predicted to be the next success in the industry. First of all, conventional ML algorithms cannot always meet the performance requirements due to the accuracy degradation caused by domain shifts. To address this issue, inductive TL [18, 19, 110–112] has started receiving more and more attention. Under the setting of inductive TL, multi-task learning is acknowledged as the most popular topic. Typically, it aims to improve the model robustness by using a small set of labeled target data set. Collecting a small set of labeled target data can decrease the training cost and enhance the robustness of the target model. The training process of inductive TL is the same as transductive TL. The only minor change is adding another target loss term to the final loss function of the model. However, the downside of inductive TL algorithms is that the training is more computationally expensive and time-consuming since another loss term is added.

Moreover, TL has been successfully applied to many applications in different fields, including but not limited to signal processing, sentiment analysis, health care system, and cyber-physical system (CPS).

Firstly, there are two main trends of signal processing, namely image processing [18, 79, 98–100], audio analysis [101–103]. Transductive TL and distant domain TL are the main streams for this field. With TL algorithms, several different real-world problems can be solved by transferring knowledge from different domains with minimized cost. Sentiment analysis has also become an extremely active field in TL, including several applications: speech recognition, recommendation system, and spam detection. For example, the study [104] proposed the first TL enabled model for language understanding. A few works contributed a lot in cross-language translation [108, 109, 119] and sentiment analysis [105–107]. Furthermore, as more attention being brought to the health system, inductive and transductive TL has also been applied to solve many health cares and medical system-related problems, such as muscle fatigue classification [113], blood test analysis [110, 111], and medical imaging diagnosis [18, 19, 112]. Especially, TL methods also benefit a number of COVID-19 related problems [120–122], such as detection, treatment, and spread prediction.

Moreover, as a newly proposed concept, CPS requires moving beyond the classical fundamental computation and physics models. Therefore, it needs new models and theories that unify perspectives, capable of expressing the interacting dynamics and integration of a system's computational and physical components in a dynamic environment. A unified science would support composition, bridge the computational versus physical notions of time and space, cope with uncertainty, and enable CPS to interoperate and evolve. Recently, there are many TL researches [114–117] conducted solid results in CPS.

2.6 Open challenges

So far, many studies of TL have carried out state-of-the results in several fields. Especially, transductive TL is the most active area in TL. However, there are still a number of open challenges of TL that are waiting to be addressed. This section discusses a number of major challenges in two levels: algorithm level and application level.

2.6.1 Challenges in Algorithms

Table 2.10: Challenges in Applications

Challenges	**Major Related Applications**
Database for TL	Social Media, Online Shopping, Browsers, Web-based Applications
Perception TL	Virtual Assistant, Smart Homes, Smart Cities, Smart Wearings, Security Systems

The author discusses several challenges at the algorithm level, such as human-guided TL, negative transfer, life-time TL, adversarial TL, and explainable TL.

First of all, most existing TL algorithms heavily rely on human instructions. Ideally, we expect models to learn an unseen task independently by using an algorithm to fully explore the data. The most successful case is AlphaZero [123] developed by Google Deepmind. It can teach itself how to master the Go game from scratch without any human experiences and instructions. However, the price of liberating the model is usually very high, and it requires a massive amount of time and computation power for the training. Therefore, the next direction is to lower the cost of this type of algorithms. In general, correctly inputting human pre-experience to the TL models can significantly reduce the time and the computation power required for training such a model. This concept is termed as human-guided TL. It aims to improve the efficiency of TL learning algorithms by correctly assembling human knowledge.

Secondly, negative transfer is widely acknowledged as an essential topic. It is one of the most significant limitations of TL. To address this issue, several distant domain TL algorithms [77, 79, 87] were proposed. Most existing methods are instance-based, and they are suffering from two major shortcomings: high case-dependence and massive source data requirement. Moreover, current methods can only transfer distant knowledge in different domains from the same modality. In other words, they can only transfer from image to image, audio to audio. Therefore, the next step of distant TL is to explore the potential of feature-based methods. Moreover, transferring knowledge between two different fields is one of the greatest challenges of distant TL, such as from image to audio and from text to image. Furthermore, an accurate domain distance measurement is also a critical factor in overcoming negative transfer. Commonly, MMD is the most popular non-parametric metric. However, it suffers from the risk of high-dimension data transformation. Other non-parametric metrics are not accurate enough for deep TL models. To address this issue, hybrid domain loss functions can help to improve the performance of distant TL.

The third, life-time TL is a relatively new concept. It aims to enable a TL framework with self-selecting the optimal learning method. The motivation behind this is that manually choosing a proper learning algorithm for a new task can be very time-consuming. Furthermore, we cannot do it manually when we are facing a new mission every time. Recently, a learning to transfer (L2T) framework [124] introduced a way to self-select an algorithm based on the input data. More importantly, there are not many studies regarding this issue. There is still a long way to go.

What is more, adversarial TL is becoming another focus in the field of TL. In general, it shares a similar idea to the original adversarial training pipeline. However, adversarial TL methods replace the feature generator with a distant feature extractor. There are a few proposed adversarial TL algorithms [31], but they are facing a critical difficulty in convergence. The convergence cannot be guaranteed in the training process due to the instability of the loss functions. Commonly, there are two counterparts in the final loss function, so the gradient explosion and disappear issues occur quite often. Therefore, designing stable loss functions will be the key to stabilize the training process for adversarial TL methods.

Furthermore, a high-level guideline for TL is also vital to the development of TL algorithms. When we develop a TL algorithm, a high-level guideline should provide comprehensive guidance to researchers in three main procedures of TL: 1) when to transfer, 2) what to transfer, and 3) how to transfer. Commonly, these three procedures can cover most high-level questions during the development of a TL algorithm. To the best of our knowledge, there are many guidance tools for conventional ML, but there is a lack of research for TL. A comprehensive guideline can help us develop algorithms and produce TL-based products in the industry.

2.6.2 Challenges in Applications

In TL applications, the author demonstrates the current challenges into four major categories: Database for TL, Perception TL, User-machine Interaction, and Job Replacement. Moreover, these challenges are related to the algorithms, policy and ethics. Primarily, the database for TL focuses on data privacy, data labeling, data cleanness, and data sharing. Perception TL is mostly related to sentiment analysis for the applications that only take speech as the input. Moreover, user-machine interaction aims to develop more user-friendly products with TL techniques. Lastly, replacing job positions with TL-enabled machines are facing many ethics issues.

2.6.2.1 Challenges in Database for TL

First of all, the database is the cornerstone of all deep learning algorithms. The database has four main challenges: data privacy, data labeling, data cleanness, and data sharing. First, data privacy means that data sets cannot be shared due to restrictions, such as the patient information of medical data, copyrights of human face data, and security requirements of aviation data. Therefore, data sets with restricted information cannot be shared to the public. Moreover, some TL algorithms involve with multiple source data sets, so they have a greater chance of violating the rules and policies. To address this issue, an extra step to filter out classified information of data sets should be added to the process of data creation. What is more, many privacy-preserving methods have been adopted to supervised learning algorithms [125]. TL can also benefit from privacy-preserving techniques [126, 127]. However, this concept has not been well investigated due to the difficulties caused by multiple data sets in different domains. Importantly, it is critical to all the applications conducted by database companies and Internet-based products.

Secondly, data labeling is another issue in TL. Unlike traditional supervised learning, TL learning does not rely on a massive amount of labeled training data, so we do not need to manually label a big data set for TL. However, many deep TL learning models require multiple data sets in different domains but with the same label space. Therefore, it creates a new challenge of labeling data sets for TL, which requires to assign domain labels to multiple source domain data sets with the same instance label space.

Moreover, it is relatively easy to discover several data sets with the same instance label space from different domains, but it is still time-consuming to manually assign domain labels when the source space is huge. In the future, creating exclusive data sets with domain labels can significantly benefit TL models. This problem is notably more critical to real-world applications with TL techniques because developing a real-world product requires way more data than academic experiments do.

2.6.2.2 Challenge in Perception TL Applications

Furthermore, perception TL concentrates on the verbal and motional inputs taken by TL algorithms, such as speech, voice, and motions. Recently, the stationary image data is considered as the most common input of most ML-enabled applications, such as auto-driving systems, smart wearings, and security systems. The easiest access is the reason why the majority of the ML-based applications most prefer the stationary image data. However, there are four major drawbacks of the stationary image input. Firstly, most existing applications are not friendly to people with disabilities. For example, stationary image-based products can cause difficulties for people who cannot type the keyboard due to their disabilities. Secondly, stationary image-based ML systems cannot easily be controlled by users. The third, ML-enabled security systems with image-based inputs suffer from safety issues because image data can be easily faked. Lastly, the domain shift can hurt the performance significantly.

To address these issues, many studies proposed to adopt other data types with less accessibility can for a wide range of applications by adopting TL techniques. For example, the study [104] proposed the first TL algorithm for speech recognition and achieved a promising performance. Furthermore, TL algorithms have been expanded to other areas: gesture recognition, voice recognition, and Micro-expression recognition. Therefore, the next stage of TL-based applications is to expand the types of input sources and enable multiple types of input sources. However, there are many unsolved problems in TL models for other types of inputs. The most challenging topic is sentiment input, such as speech and text. For example, most products can only take keywords as inputs but cannot handle longer sentences. There are many successful TL algorithms for image processing tasks, but there are not many studies in sentiment analysis. Recently, some works [105, 128] have introduced TL algorithms for sentiment-focused long speech analysis. Therefore, adapting TL techniques to real- world products with perception inputs is a very challenging topic.

2.7 Concluding Remarks

Finally, the number of TL-related researches has been on a rapid increase in the past decade. Moreover, its usage in industries is bypassing supervised learning due to its advantages on efficiency and performance. In the future, with the above four main challenges being addressed, TL will be more widely used in both academia and industry.

Chapter 3

Data-Efficient Machine Learning Framework and An Application Case

In this chapter, a novel DEML framework and a DEML evaluation framework will be introduced. Particularly, it is a software-oriented and product-focused framework that is designed for DEML. However, it can be generalized to other conventional ML applications with a few small adjustments. And then. the author will introduce an real-world application with TL.

3.1 Data-Efficient Machine Learning Framework

As a sub-field of artificial intelligence, Machine Learning (ML) was proposed decades ago, and it is now attracting more and more attention. In the beginning, machine learning was not preferred by most researchers because of its poor performance, which was limited by insufficient data and weak computational power. To solve these two main issues, many powerful processors (TPU and GPU) have been built to give us the ability to train extremely complicated models, and the internet has made data more accessible. Recently, with more accessible data and powerful machines, the performance of machine learning models has been brought to a whole new level, such as [10, 11]. However, in many real-world problems, either we do not have adequate well-labeled training data or do not have enough unlabeled data to train an accurate model for a specific task even though the data is much more accessible than it was before. In addition, even with a sufficient amount of data available, the training process for deep models can be too costly. As these problems become the new challenge, Data-Efficient Machine Learning (DEML) has been proposed to improve the efficiency and the performance of ML. The goal of DEML is to enable us to build models with an insufficient amount of data or limited computation power.

As shown in Figure-3.1, DEML covers a wide range of topics in data science and learning algorithms, which can provide a scientific guideline to deal with insufficient training data and incompatible computational power in modern ML. In general, there are three main components: data science, learning algorithms, and generative adversarial networks (GANs). Moreover, GANs is a type of Deep Neural Nerworks (DNNs) commonly used in learning algorithms and data augmentation. DEML framework benefits many tasks, such as computer vision, NLP, and data analysis. For data science, the goal is to expand the volume of training data sets artificially. For learning algorithms, the goal is to reduce the reliance on massive data by alternating the architectures. This section will provide comprehensive overviews of each component in the DEML framework and explain how each component can help us deal with insufficient training data and incompatible computational power.

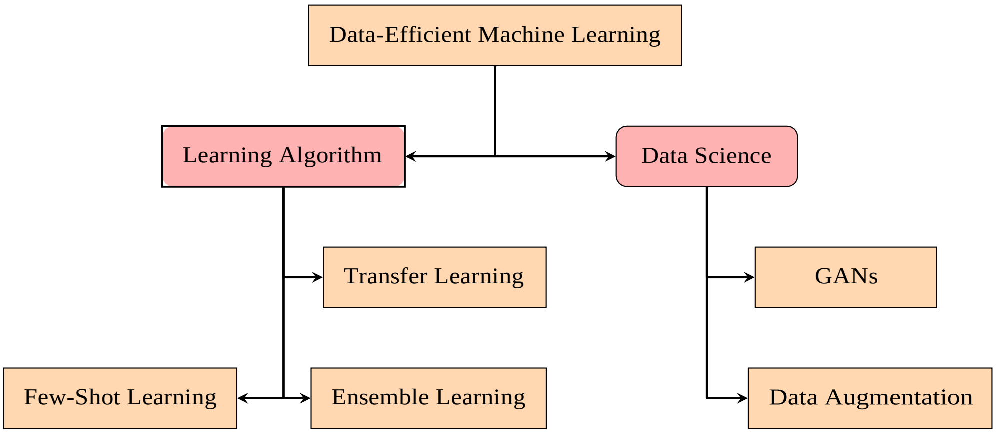


Figure 3.1: Data-Efficient Machine Learning

3.1.1 Data Science

Training a deep ML model is repeatedly tuning a large set of hyper-parameters to optimize the final performance. The volume of required training data is proportional to the size of the model. Generally, there are two ways to solve this issue: 1) reduce the number of hyper-parameters and 2) obtain more training data. From the perspective of data science, the focus is on obtaining more training data.

Data science has played an essential role in data-efficient machine learning methods. It is not only for data analysis but also for solving the issue of insufficient training data. In general, data science methods are frequently used for deep machine learning methods that require a significant amount of training data. This section categorizes data science methods into two main sub-fields, data augmentation, and data re-sampling.

3.1.2 Data Augmentation

In machine learning, data augmentation is widely acknowledged as an effective way to address insufficient training data. And, data augmentation has been beneficial to many practical problems in different areas, such as image processing [129–132], audio

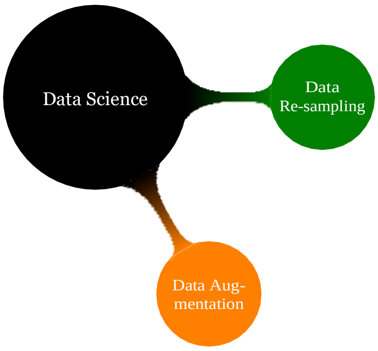


Figure 3.2: Mindmap of Data Science

analysis [133–135], and signal processing [135, 136]. In theory, we expect a well-trained model to be robust under many different situations, such as different backgrounds and different angles. Commonly, a massive amount of training data is usually required for the model to learn the invariant features. However, manually collecting training samples from different orientations and backgrounds is very time-consuming. Thus, we wish to generate new samples that contain diverse distribution by using data augmentation methods. In this subsection, the author mainly focuses on data augmentation in the following three fields, image processing, audio analysis, and signal processing, respectively.

Table 3.1: Data Augmentation

Data Augmentation	
Image Processing	[10, 76, 129–132, 137–139]
Audio Analysis	[133, 134]
Signal Processing	[1, 49–51]

3.1.2.1 Data Augmentation for Image Processing

Firstly, [132] has briefly introduced many state-of-art image data augmentation techniques for deep learning. As shown in Figure-3.3 , it is categorized into two main branches, basic image manipulations, and deep learning approaches. In many image processing tasks, kennel filters, geometry, and color space transformations, random erasing, and mixing images are used most commonly. Moreover, deep learning methods are also frequently applied when conventional methods do not work well.

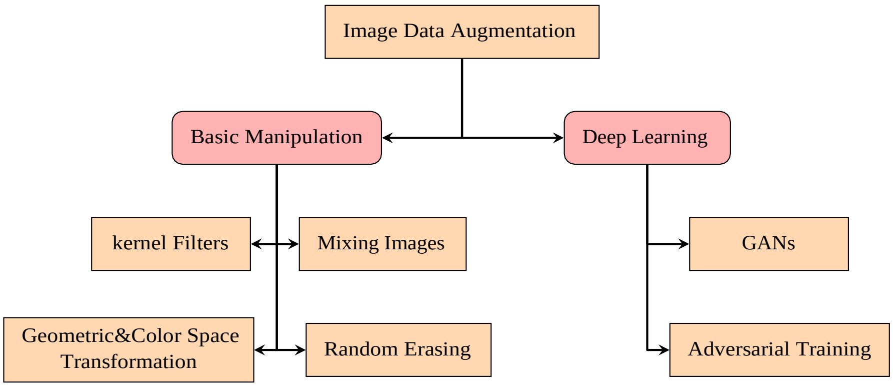


Figure 3.3: Image Data Augmentation

For basic manipulation, geometric and color space transformations are applied to numerous of deep neural networks, such as Resnet [76], LeNet-5 [137], and AlexNet [10]. Moreover, the kernel filter method is one of the most popular ones which can sharpen and blur images by applying sliding filters to images. [139] introduced a kernel filter called PatchShuffle that randomly swap pixel values in the filter. Mixing images is also an effective solution for obtaining new data from the existing data. Firstly, [140] introduced a novel data augmentation method called Between-Class learning (BC learning). In the first place, it was inspired by [141], a data augmentation method designed for sound recognition tasks. Additionally, random erasing [142] is also another common image data augmentation technique. It randomly alters the pixel values of certain areas of an image. Typically, there are few regular ways to alter the pixel values, such as, filling the areas with the mean value, 0, 255, or random values. Unlike other methods, random erasing mainly focuses on decreasing over-fitting, while it can also deal with the issue of lacking training data.

3.1.2.2 Data Augmentation for Audio Analysis and Signal Processing

Signal processing and audio processing share several common behaviors, so the author merges them into one section in this survey. Commonly, alternating the length of signals is considered the most common data augmentation technique for signal processing. For example, a vocal tract length (VTL) method [143, 144] can produce three alternatives by scaling the original audio with three speed factors, 0.9, 1.0, and 1.1. Secondly, [145] proposes a modified vocal tract length perturbation (VTLP) method, which applies a deterministic perturbation factor, α,

$$\alpha \rightarrow \{\alpha \pm \Delta, \ldots, \alpha \pm k\Delta, \ldots, \alpha \pm K\Delta\}, k = 1, \ldots, K \qquad (3.1)$$

where, $2K$ is the total number of replicas of the original data and δ is a fixed shift along the α axis. Moreover, stochastic feature mapping (SFM) augments training samples by statistically converting one speaker to another. In addition, SFM seeks to create a mapping from the source speaker $O^{(S)}$ to the target speaker $O^{(T)}$ when both speakers speak the same utterance u with label L. As the equations are shown below:

$$O^{(S)} = \Big\{ o_1^{(S)}, \ldots, o_N^{(S)} \qquad o_t^{(S)} \in H$$

$$O^{(T)} = \Big\{ o_1^{(T)}, \ldots, o_N^{(T)} \qquad o_t^{(T)} \in H$$

$$\hat{F} = \underset{F}{Argmin}\,\Gamma \Big(F(\mathbf{O}^{(\mathbf{S})}), \lambda_{\mathbf{H}}^{(\mathbf{T})} \Big)$$

where $\lambda_H^{(T)}$ represents the acoustic model of the target speaker $O^{(S)}$ in the feature space H and then estimates a transformation F to minimize a chosen objective function Γ. In addition, [146, 147] introduce synthesis speech data created by concatenating existing waveform segments with statistical approach, like Hidden Markov Models

(HMM).

3.1.3 Data Re-Sampling

Importantly, generating extra new samples from existing data sets usually does not create different distributions. In other words, it cannot always enhance the robustness of a model. And, augmentation is not the only way to improve the performance of a model when the training data is insufficient. Re-sampling methods can achieve the same goal without generating new data points, and they are always recognized as an indispensable tool for machine learning. Generally, they involve repeatedly drawing samples from a training set and refitting a model of interest on each sample to obtain additional information about the fitted model.

However, re-sampling methods are usually computationally expensive because they require fitting the same statistical method multiple times using different training data subsets. However, due to the recent advances in computing power, the computational requirements of re-sampling methods are not prohibiting. In this section, the author discusses two of the most commonly used methods, validation & cross-validation and bootstrap.

3.1.3.1 Validation and Cross-Validation

In the absence of an extensive designated test set that can be used to estimate the test error rate directly, a number of techniques can be used to estimate the quantity using the available training data. Most commonly, we can apply the validation set approach for deep learning models by simply separating the training into training and validation sets based on a certain ratio. However, taking away a subset from the validation training is always feasible when we only have a small set of training data. Therefore, we tend to perform cross-validation for relatively small data sets, which shares many commons with the validation set approach.

Firstly, assume that we try to estimate the test error associated with fitting a deep learning model to a particular distribution. We have a normal size data set that is big enough to let us perform the validation set approach. Then, we can randomly divide the training set into two parts, a training set and a validation set. The model then fits the training set, and the fitted model is used to predict the validation set. The errors of prediction results on the validation set can be assessed using a MSE (Mean Square Error). The error rate can then be back-propagated to the model and make further adjustments on parameters to get better performance. The validation set approach is conceptually simple and easy to apply, but it has three main drawbacks:

- The validation estimate of the test error rate can be highly variable, depending on precisely which observations are included in the training set and which are included in validation.
- Only a subset of observations - those are included in the training set rather than in the validation set - are used to fit the model. Then the model tends to be over-estimated on the test set error rate since the model was fitted on fewer samples.
- It requires a relatively big data set, so it is not feasible when we cannot afford taking a subset out of the given training set.

To address the above issues, the cross-validation approach is considered as an effective solution. Leave-One-Out Cross-Validation (LOOCV) is the most common method closely related to the validation set approach. Similarly, LOOCV also involves splitting the training set into two parts. Differently, instead of creating two subsets of comparable size, single observation (x_1, y_1) is used for the validation set, and the remaining observations $(x_2, y_3)...(x_n, y_n)$ make up the training set. The model is fit on the $n - 1$ samples, and a prediction $\hat{y}_1$is made for the excluded observation. And more, $(y_1 - \hat{y}_1)^2$ provides an approximately unbiased test error. However, it is a poor estimate because it is highly variable since it is based on a single observation. The LOOCV estimate for the test error is the average of n samples:

$$CV_{(n)} = \frac{1}{n} \mathop{}_{n}^{i=1} MSE_i \tag{3.2}$$

As a shortcoming of LOOCV, it can be too costly to implement since the model has to fit the data n times. Especially for deep learning models, this can be very time-consuming with a big set of data. However, it is an efficient method for simple models, such as polynomial regression.

3.1.3.2 Bootstrap

Bootstrap is an alternative data re-sampling method that estimates quantities about a set of data by averaging estimates from multiple small subsets of the data. Unlike validation and cross-validation, bootstrap can be presented with confidence intervals. Moreover, a classic bootstrap procedure can be summarized as follow:

- Initialize the number of bootstrap samples and the sample size.

- For each bootstrap sample, perform: 1) draw a sample with replacement with the chosen size; 2) fit a model on the data sample; 3) estimate the skill of the model on the out-of-bag sample.
- Calculate the mean of the sample of model skill estimates.

3.1.4 Learning Algorithm

As mentioned previously, the reliance on big data is decided by the design of learning algorithms. From this point of view, the focus is to reduce big data demand by alternating model architectures. In the DEML framework, there are five main learning disciplines: TL, non-parametric learning, few-shots learning, and ensemble learning. Moreover, TL is the major concentration in this dissertation.

3.1.4.1 Transfer Learning

Traditional ML relies on a massive amount of training data. It assumes one critical condition: the training data and the testing data are drawn from the exact same distribution. However, this assumption does not always hold in many real-world problems. As such, most conventional ML algorithms usually suffer from three main difficulties: insufficient data, incompatible computation power, and distribution mismatch. First of all, various solutions have been proposed to address the first two problems, such as data argumentation, data synthesis, distributed learning, and cloud computing. However, each of these proposed solutions suffers from some adversities, such as regarding cost, efficiency, and security. Recently, transfer learning (TL) has been brought to our attention to deal with all three difficulties.

Primarily, TL aims to solve the target task by leveraging the knowledge learned from source tasks in different domains, so it does not need to learn from scratch with a massive amount of data [23, 26, 27]. As such, TL first can address the most significant issue, insufficient well-labeled training data. Moreover, the time and computation resources required for training a model can also be greatly decreased since pre-learned knowledge from other domains and tasks can be reused. Furthermore, the distribution mismatch can cause significant performance degradation on ML models. TL can also address it by fusing knowledge from one or multiple different domains. The rest of this dissertation will introduce a review of TL, novel TL algorithms, and cutting-edge TL applications.

3.1.4.2 Few-Shot Learning

Few-Shot Learning (FSL) is counter-intuitive to the conventional ML concept. It aims to develop a robust model with very few training samples or no training samples. In general, FSL has several sub-fields, such as one-shot learning (OSL), N-shot learning (NSL), and zero-shot learning (ZSL). Moreover, the inspiration of FSL is more similar to human nature. For example, a person can recognize an unseen landscape if he has adequate information about its appearance, properties, and functionalities. The information can be learned from other sources, such as books, the internet, and radios. In this scenario, the person can learn how to recognize an object without seeing it or its images in advance. Therefore, a ML model can learn rare classes using FSL techniques, and the training cost can be greatly reduced. There are several common FSL algorithms: Model-Agnostic Meta-Learning (MAML) [148], Matching Networks [149], and Prototypical Networks [150].

There are three common FSL approaches: 1) data-level approach, 2) parameter-level approach, and 3) meta-learning approach.

In data-level approach, FSL aims to solve a task with insufficient training data by using knowledge from other large base-data sets. In addition, the base-data set does not have the classes that we have in our support-set for the FSL task. Besides, data augmentation and GANs can be used to increase the volume of the training data artificially. However, the data-level approach can lead to the over-fitting issue. Therefore, the parameter-level approach is used to overcome this disadvantage. It usually limits the parameter space and uses regularization and proper loss functions. The model will generalize the limited number of training samples. Moreover, it can enhance model performance by directing it to the extensive parameter space. Furthermore, in the meta-learning approach, a model is learning to learn if its performance at each task improves with experience and the number of tasks. Meta-learning approach learns common features shared by the target and the base sets instead of learning the target objects directly. In general, there are two main types meta-learning approaches: metric-learning and gradient-based learning. Metric-learning algorithms learn to compare data samples. In the case of a Few-Shot classification problem, they classify query samples based on their similarity to the support samples. In image processing tasks, it trains a convolutional neural network to output an image embedding vector, which is later compared to other embeddings to predict the class. Differently, gradient-Based approach, you need to build a meta-learner and a base-learner. A meta-learner is a model that learns across episodes, whereas a base-learner is a model that is initialized and trained inside each episode by the meta-learner.

3.1.4.3 Ensemble Learning

First of all, ensemble learning aims to create a strong learner by combining two or more weak learners. Generally, ensemble learning has three main advantages of: 1) weak learners do not require a massive training data set, 2) it improves flexibility and can scale in proportion to the volume of training data, and 3) it can boost the robustness and the performance. Moreover, the weak learners that contribute to the strong learners can be either the same type or different types. They can even be trained with different data sets depending on the specific situations.

In real-world problems, unbalanced data sets can greatly benefit from ensemble learning. The classes with insufficient training samples can be trained on simple learners, such as tree-based models, yield decent results with fewer data. Differently, other classes can be trained with computational-expensive models, such as neural networks. And then, predictions made by the ensemble members may be combined using statistics, such as the mode or mean, or by more sophisticated methods that learn how much to trust each member and under what conditions.

Commonly, ensemble learning has two major methods: bagging and boosting. Bagging trains a bunch of individual models in a parallel way. A random subset of the data trains each model. Boosting trains a bunch of individual models in a sequential way. Each individual model learns from mistakes made by the previous model. Moreover, bagging can decrease variance, and boosting can decrease bias. More importantly, besides addressing insufficient training data, ensemble learning can also handle heterogeneous learning tasks. Furthermore, there are two commonly used ensemble learning methods: Adaboost [60] and random forest [151].

3.2 Product-focused DEML Evaluation Framework

In this section, a product-focused DEML evaluation framework will be introduced. First of all, this framework is software-oriented and product-focused. There is a significant difference between the proposed evaluation framework and common experimental and theoretical evaluation strategies used by most ML competitions. In general, most researches only concentrates on the situation that both training and testing data are from the same distribution. In other words, the given training set and testing set are just two subsets that are randomly split from the whole data set. Therefore, the performance degradation on the testing set is not caused by distribution. However, in real-world problems, the testing set is usually collected from domains that are different from the training set domain. The distributions of the source domain (training set) and the target domain (testing set) are more or less different. For example, a car detection model trained on a set of images collected from Orlando might not work well on the testing images collected from Beijing because the difference between the two cities is significant. Thus, the distribution mismatch is an essential factor that can lead to serious accuracy reduction.

From this aspect, the proposed framework aims to improve the performance of DEML in the industry by paying close attention to the potential loss related to distribution mismatch. Moreover, this framework also provides a clear and efficient guideline to develop DEML models for practical problems. Furthermore, this framework is not only profitable to DEML models but also conventional ML models. According to the particular goal of the given task, a few simple adjustments can make this framework to fit any ML problems. In the following sections, a number of notions and terms will be introduced first. And then, the details of the framework will be discussed.

Importantly, there are several different types of model performances:

- $Performance_H$: Human-level performance. It is the average performance can be achieved by human. It is the benchmark for the DEML model.
- $Performance_T$: Training performance. Model performance on the training data. It is the highest accuracy that a DEML model can produce on the training set.
- $Performance_{TD}$: Training-development performance. Model performance on the validation data that is under the same distribution as the training data.
- $Performance_{T\ est}$: Testing performance. Model performance on a small set data collected from the real-world, which is under the different distribution as the training data.
- $Performance_D$: Development Performance. Model performance on a bigger set of real-world data, which is under the same distribution as the testing data.

Moreover, there are another few terms for different types of performance reductions:

- $Bias$: the performance reduction between $Performance_H$ and $Performance_T$.
- $V\ ariance$: the performance reduction between $Performance_T$ and $Performance_{TD}$.
- $Distribution\ Mismatch$: it is reflected by the performance reduction between $Performance_{T\ D}$ and $Performance_{Test}$.
- $OverfitRate$: It is reflected by the performance reduction between $Performance_{T\ est}$ and $Performance_D$.

Moreover, the main goal is to show more details of the evaluation framework and explain how it can be helpful to develop a DEML model. What is more, the generalization of the proposed framework will also be justified.

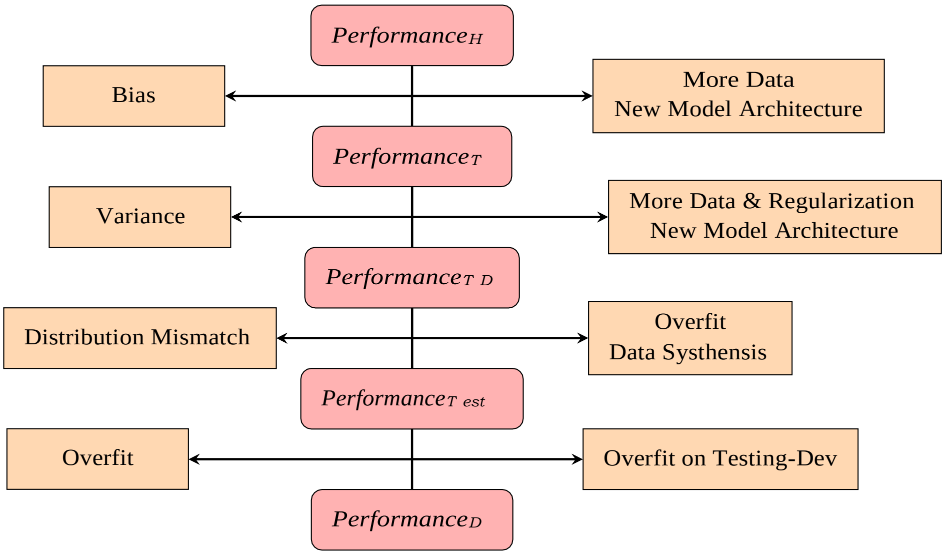


Figure 3.4: DEML Evaluation Framework

As shown in Figure-3.4, the framework is built upon five types of performances as mentioned in the previous section. With those performances, four different losses can be measured, and adjustments of the model can be made. To build a robust DEML model, there are five main steps:

- Step 1: produce preliminary research to justify the best and average human-level performance of the task. It is only reasonable to replace human by machines if the machine can perform at the same level or even better than human can do.
- Step 2: perform data analysis and choose a proper algorithm. Adjust learning algorithms or add more data if the value of the *bias* is high. Moving to next step without lowering the *bias* will cause greater performance decrease in the future steps.

- Step 3: run the model on the validation set that is under the same distribution as the training data. Low *variance* proves that the model is not overfit on the training data, then it is ready for the next step. There are two common reasons that can lead to a high *variance*, 1) insufficient training data; 2) need regularization term. Commonly, data augmentation and DEML methods can be used to dramatically reduce it.

- Step 4: run the model on a small set of real-world data. The performance decrease is usually caused by distribution mismatch. To close the mismatch, data synthesis and transfer learning can be used.

- Step 5: finally, test the model on the practical problem. If the performance degrades significantly, it means the model is overfit on the development data. Applying regularization techniques and data augmentation can make the model more robust.

3.3 Transfer Learning-based Waste Sorting

We are entering a new era of smart cities, which offers great promise for improved wellbeing and prosperity but poses significant challenges [152–154]. Machine learning and data analytics have emerged as essential tools to address these challenges, which smart cities are facing [155–158].

Rapidly increasing pollution from overpopulation and industrialization is causing serious damage to the natural environment of the Earth. As the consequences, water pollution, air pollution, and deforestation are causing a number of negative effects on our health and the economy, such as the increasing cancer rate, new diseases, extinction of species, and soil contamination. For example, toxic materials can be transferred into human bodies and wildlife from air, water, and food. Moreover, soil contamination can seriously hurt all fields related to agriculture. As shown in the study of [159], the expense of pollution control has been exponentially increasing in the past few decades, and many potential solutions have been proposed. To the best of our knowledge, recycling is widely acknowledged as one of the proven ways to reduce environmental pollution effectively. In general, the benefits of cycling are listed as follows: reducing the waste lost in landfills, reducing greenhouse gas emissions, and saving resources for making raw materials. Furthermore, accurately sorting the waste from our daily life is the first and very important step of the big picture of recycling. Therefore, finding an effective and efficient way is the key to the success of the cycling process.

In this chapter, our focus is on building a DL model for solid waste sorting, which lands in the field of image classification. Firstly, traditional image processing methods use hand-designed features to complete tasks like classification, detection, segmentation. However, designing features by hand is a very time-consuming and costly process. Furthermore, it does not always output promising performance in complicated tasks. In the recent decade, DL has dominated this field by dramatically setting our hands free from designing features, and improving the performance. Additionally, one of the most famous DL models, convolutional neural network (CNN), has shown its great power in a number of different fields, such as object classification, object detection, and speech reorganization. Generally, a deep neural network (DNN) tends to enable the machine to learn how to accomplish the task. In other words, DNN can be considered as a black box of a massive amount of hyper-parameters. The goal is to get the best performance by iteratively adjusting the values of parameters based on a set of rules. However, most DL methods require a huge set of well-labeled training data to get promising performance. In many real-world problems, we do not have a sufficient amount of labeled data for training, or we cannot even find unlabeled training data. Researchers started focusing on transfer learning to address this issue, which allows us to leverage the knowledge stored in other well-trained models. Moreover, we do not have many datasets for waste sorting tasks that can provide enough training data for deep networks. Therefore, the author proposes a transfer learning model for this topic.

According to [23], there are three common transfer learning settings: inductive transfer learning, transductive transfer learning, and unsupervised transfer learning. In general, there are multiple different domains in a transfer learning task: one target domain and one or multiple source domains. As for inductive transfer learning, supervised training data is always available in the target domain. In the setting of transductive transfer learning, the well-labeled data is only available in the source domain. Differently, there is no labeled data in both the source domain and the target domain in the setting of unsupervised transfer learning. In this review, the setting of the proposed model fits into inductive transfer learning. In addition, there is only a small set of data [160] that contains 2530 images in total, which might not be enough for building a robust waste sorting model. the author tends to use domain adaption techniques to leverage the knowledge stored in deeply-trained models like, AlexNet [10], ResNet [161], that are trained on ImageNet dataset. By doing so, the author was able to push the testing accuracy to 96% by using such a small dataset.

The rest of the chapter is organized as follows. Section 3.3.1 presents related work. Dataset is introduced in Section 3.3.2. the author presents the proposed methodologies in Section 3.3.3. Moreover, experimental results are discussed in Section 3.3.5. Section 3.3.6 gives a conclusion.

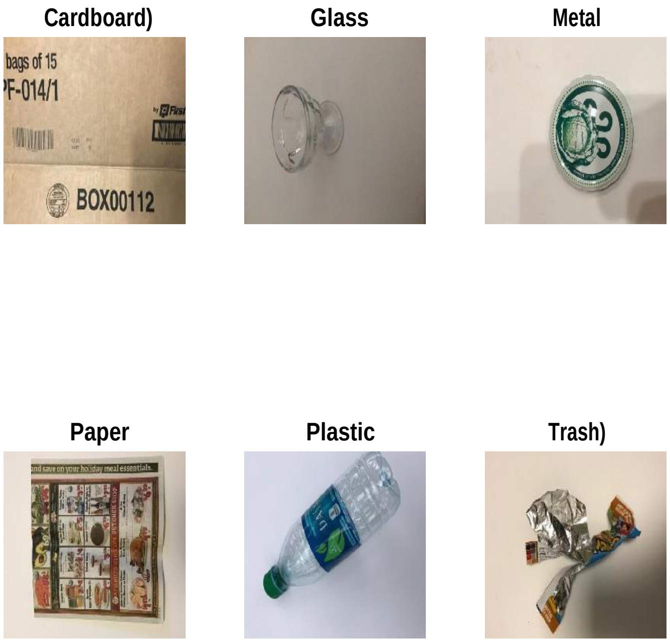


Figure 3.5: Source Data & Target Data.

3.3.1 Related Work

Previously, many image classification projects have been created. However, there are not many that are related to waste sorting. In this section, the author introduces a number of projects that are related to waste sorting. Moreover, for a better understanding, the author categorizes them into three sub-fields: traditional methods, conventional DL methods, transfer learning methods.

3.3.1.1 Traditional Methods

Firstly, a traditional model, support vector machine (SVM), is considered one of the best initial image classification methods. Moreover, comparing to DL models, it is simpler to build and easier to train. [160] built an SVM model for waste sorting based on a hand-designed feature detector, SIFT. In addition, the SIFT descriptor is one of the most powerful feature detectors, and it is invariant to scale, noise, and illumination [162]. Thus, it is extremely helpful to waste sorting. Furthermore, the best kernel of SVM was found after testing a number of different kernels. It is defined as:

$$K(x,x^{'}) = exp(-\frac{\|x - x^{'}\|^2}{2\sigma^2}) \tag{3.3}$$

And, the best performance achieved by SMV was 63% testing accuracy.

3.3.1.2 Conventional DL Methods

Importantly, as mentioned in the earlier contents, one of DL methods' greatest advantages is that deep networks can automatically learn features, instead of designing features by hands. However, DL models require matching the size of data and the size of the network. A significant mismatch usually causes over-fitting or under-fitting. [160] built a CNN that is considered as a simplified version of AlexNet [10]. As claimed by the authors, this model only achieved 22% testing accuracy, which is worse than a pure guess. Moreover, [163] selected three successful DL architectures, namely, MobileNet [164], DenseNets [165], and Inception [166], to train from scratch. As a result, those models achieved testing accuracies, 84%, 84%, and 89%, respectively. DL models achieve better performance than traditional models.

However, there are two main drawbacks of conventional DL methods. Firstly, those selected models are reasonably deep and complicated. Training from scratch is very time-consuming and can be over-fitting with such a small dataset. Secondly, one advantage the proposed model has is that there are a number of datasets that contain the objects that are in TrashNet. Furthermore, it can benefit from those samples in other datasets if distribution mismatches can be reduced. However, conventional DL methods cannot take advantage of those samples from other domains.

3.3.1.3 Transfer Learning Model

To address drawbacks of conventional methods, numerous transfer learning methods have been proposed. Commonly, the distribution mismatches between the source domain and the target domain are the main issue that prevents us from using samples collected from different domains for training. As one of the solutions, fine-tuning is acknowledged to be an effective way to deal with the distribution mismatch. Primarily, [163] also implemented fine-tuning on the selected DL architectures to improve the performance to a new level. As shown in Table-3.2, the authors pushed the best testing accuracy to 95% by combining fine-tuning and data argumentation. Fine-tuning not only produces a better testing accuracy but also dramatically reduces the training time.

Moreover, the author would also like to expand the dataset by leveraging the samples collected from other domains. In this study, the author implements transfer learning methods DDC [2], DeepCoral [17], to push the performance to an even higher level. Generally, TL methods tend to reduce the distribution mismatch by adding an additional constraint term to the loss function. For example, DDC deploys Maximum Mean Discrepancy [167] (MMD) and DeepCoral use CoralLoss to measure the distance between two domains so that the mismatch can be reduced. For our models, the author modifies the original loss functions in the original paper of DDC and DeepCoral. Finally, the best testing accuracy, 96%, was achieved by DeepCoral-based model.

Table 3.2: Performance Overview

Methods	Testing Accuracy
Traditional Methods	63%
Conventional DL Methods	22%
Transfer Learning Methods	95%
Ours	96%

3.3.2 Dataset

Firstly, there are not many open-source datasets for waste sorting. One of them, the TrashNet [160] was collected by students in Standford, which contains six classes: paper, glass, metal, cardboard, plastic, and trash. There are 2527 images with white background, and there are all resized to 512 by 384. Moreover, a few samples of each class of TrashNet are demonstrated in Figure3.5.

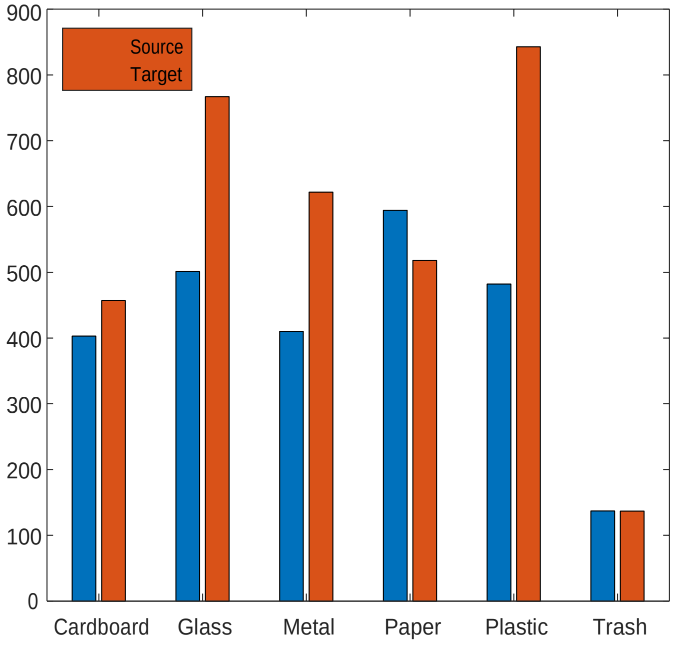


Figure 3.6: Source Data & Target Data.

Importantly, this is a fairly small dataset that might not be able to train a model with high-accuracy. And, [160, 163] all used data augmentation techniques to expand the dataset. However, objects in TrashNet are all very common things and can be easily found in other datasets but with different distributions. In this study, the author wishes to benefit from the datasets in other domains using transfer learning techniques to deal with the distribution mismatch. In addition, there is another dataset [168] that has collected from different distributions but contains very similar objects as TrashNet, so that it can be used as the source data. Moreover, the distributions of the source data and the target data are shown in Figure3.6. As we can tell from the figure, the distribution of the sample number of each class is imbalanced. Therefore, the author first balanced out the sample number of each class by applying basic image data augmentation, such as flip, rotation, kernel filters.

3.3.3 Methodology

As mentioned earlier, conventional DL algorithms have two significant shortcomings: insufficient training data and domain shift. Moreover, these two drawbacks significantly limit the potential of DL being applied to waste sorting. To address this problem, the author proposes to adopt transfer learning to develop a robust waste classification model with a limited amount of training data.

As a sub-field of data-efficient learning algorithms, transfer learning is currently one of the most popular topics. The concept of transfer learning is to solve the target task by leveraging the knowledge learned from source tasks in different domains, instead of learning from scratch and requiring massive data. Generally, traditional machine learning algorithms assume that training and testing data are in the same feature space and share the identical distribution. However, this assumption does not always hold in many real-world problems [25, 54–56]. One example is Office31 [169] classification, where we have a precise model trained on tons of data collected by webcam, but we now want to build another model using a small amount of data collected from Amazon. In this case, the author wishes to generalize the knowledge learned from the source domain to the target task with a completely different distribution. For this kind of problem, transfer learning can deal with the limited data issue and significantly reduce the time for training.

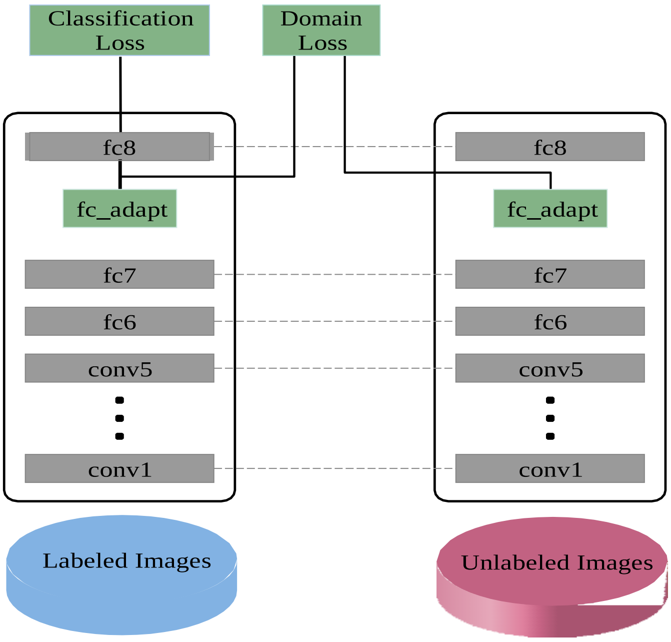


Figure 3.7: Deep Domain Confusion.

As introduced by [23], there are three categories of transfer learning, inductive transfer learning: transductive transfer learning, and unsupervised transfer learning. In this research, waste sorting is similar to multi-task learning problem, which lands into the setting of inductive transfer learning. For inductive transfer learning, the source domain and the target domain usually have labeled data in both domains. However, the target domain's training data is not always enough, so we need to transfer the knowledge learned from the source domain. This study implemented a novel loss function with dynamic weighting and built four different models, DDC-AlexNet, DDC-ResNet, DeepCoral-AlexNet, and DeepCoral-Resnet.

3.3.3.1 DDC-AlexNet

Previously, Alexnet [10] won the ILSVRC02012 competition and achieved top-5 test error rate of 15.3% on the ImageNet data-set. Firstly, the idea of the adaptation layer was proposed by [39]. It introduced a modified feedforward neural network, Domain Adaptive Neural Network (DaNN), with one adaptation layer. Importantly, the loss

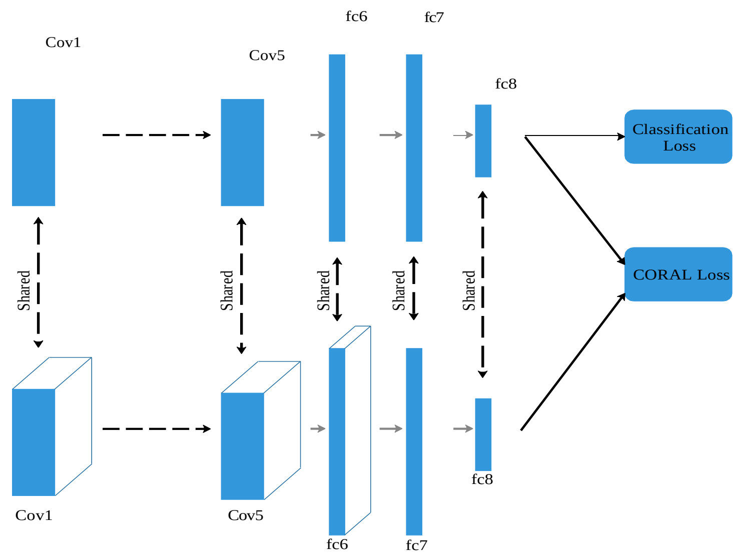


Figure 3.8: Deep coral with AlexNet backend.

function is constructed by two parts, the general loss, and the MMD regularizer, respectively. Additionally, the MMD loss is used to evaluate the distribution mismatch between the source and target domains. However, it is a very shallow and simple model, so the performance is still limited. To achieve better performance, the author wishes to extend the potential of DaNN to deeper networks. As illustrated in Figure3.7, Deep Domain Confusion (DDC) [2], an AlexNet-based [10] Convolutional Neural Network (CNN) with one adaptation layer was proposed to learn a semantically meaningful and domain invariant representation. Additionally, the evaluation metric can also be used to determine the position and the dimensionality of the adaptation layer.

Additionally, DDC deploys a loss function that contains two terms, classification loss L_C , and MMD constraint MMD^2. As shown in (3.4), X_S and X_T represent the data sets from the source domain and the target domain. Moreover, λ determines how strongly the author would like to confuse the domains.

$$L = L_C(X_L, y) + \lambda MMD^2(X_S, X_T) \tag{3.4}$$

$$MMD(X_S, X_T) = \left\| \sum_{i=1}^{n_S} \varphi(x_S^i) - \sum_{j=1}^{n_T} \varphi(x_T^j) \right\|_H^2 \tag{3.5}$$

In addition, λ is a fixed coefficient, as described in the original paper. However, setting a reasonable value to it is not a simple process. Greater value can lead the model to focus too much on reducing the distribution mismatch, while smaller value might get poor classification accuracy on the target domain due to not focusing enough on the distribution mismatch. Therefore, the author proposed to make λ to be a dynamic factor. As described in (3.6), it is a hyperbolic-tan function that scales from 0-1. Theoretically, we wish to focus on extracting domain-invariant features in the early stage and shift the focus on enhancing the target classification accuracy at the later stage.

$$\lambda = tanh(0.02x) \tag{3.6}$$

3.3.4 DDC-ResNet

Moreover, DDC is transfer learning architecture that can be easily generalized to other pre-trained DL models. In this study, the author also examined ResNet-based DDC model. However, the adaption layer with dynamic loss function is added after the last average-pooling layer.

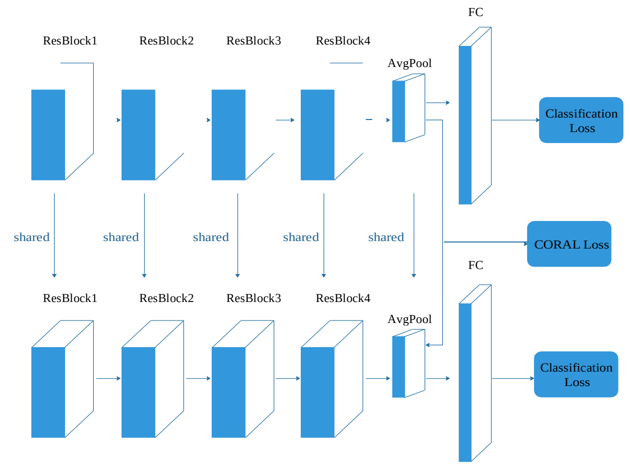


Figure 3.9: Deep coral with Resnet backend.

3.3.4.1 DeepCoral-AlexNet

Furthermore, [165] introduced another transfer learning framework, DeepCoral, which shares a similar idea as DDC. As shown in Figure3.7 , it places one adaption layer after the last fully connected layer with a new loss function, CoralLoss. f_{CORAL}, is defined as the distance between the second-order covariances of the source and the target features. And, it is described in (3.7),

$$f_{CORAL} = \frac{1}{4d^2} \| C_S - C_T \|_F^2 \tag{3.7}$$

where C_S and C_T are feature covariance matrices, $\| \cdot \|^2$ is the squared matrix Frobenius norm. Moreover, inspired by multi-kernel MMD [170], the author first proposes a novel distribution distance measurement, Dual Dynamic Domain Distance (4D). As demonstrated in (3.8), 4D domain loss combines two different evaluation metrics since a single metric might not be good enough for an accurate domain distance measurement.

$$4D = \frac{1}{2}(f_{MMD} + f_{CORAL}) \tag{3.8}$$

Finally, the author dynamically combines the classification loss f_{Class} and domain loss $4D$ as the final loss function:

$$f = f_{Class} + \lambda 4D \tag{3.9}$$

3.3.4.2 DeepCoral-ResNet

Same as DDC, DeepCoral also can be generalized to other pre-trained networks. As shown in Figure3.9, the author extended it Resnet by adding the adaption layer after the last average-pooling layer.

3.3.5 Experimental Results

3.3.5.1 Experimental Setup

As mentioned in section 3.3.2, there are 2754 labeled-images in the source domain, and 2530 labeled-images in the target domain. In addition, images in two domains have the same set of labels but different distributions. In the experiment, the author split the target dataset into Target train and Target test by the ratio of 80/20. Moreover, the total epoch is set to 200. Additionally, to extend the dataset even further, the author also applied simple data augmentation techniques to both the source data and the target data. Specifically, horizontal flipping, small rotation, and adding Gaussian noise were performed.

3.3.5.2 Results

According to Table 3.3, comparing to other existing models built on TrashNet, our transfer learning models achieve better performances in general and DeepCoral ResNet with novel 4*D* loss has achieved the best testing accuracy, 96% with 75 epochs. Moreover, the only previous model that is close to DeepCoral ResNet is the fine-tuned DenseNet model. What is more, we can see from the Table3.3 is that transfer learning models are all the better than traditional models and conventional DL models.

Table 3.3: Transfer Learning Performance

Models TL	**Epoch**	**Testing Accuracy**
DeepCoral_ResNet	75	96%
DeepCoral_AlexNet	80	93%
DDC_ResNet	85	95%
DDC_AlexNet	75	93%
DenseNet_ Fine-tune	120	95%
Models Not TL	**Epoch**	**Testing Accuracy**
SVM	100	63%
Inception-V1	100	89%

Furthermore, in all models built by us, DeepCoral ResNet gives the best performance, 96% testing accuracy. Additionally, as plotted in Figure3.10, ResNet-based models are generally more accurate than AlexNet-based models. As shown in the figure, all four models converge around 60 - 80 epochs, which is considerably faster than the fine-tuning models proposed in [163]. However, TrashNet is still relatively small for the DL architectures like ResNet, and AlexNet. The performances of the AlexNet-based model start dropping after 130 epochs. Furthermore, the models start over-fitting from there. Differently, ResNet-based models maintain stable through all 200 epochs.

To show that the 4D loss function can improve the performance, the author made a comparison between DeepCoral ResNet with regular loss function and the same model

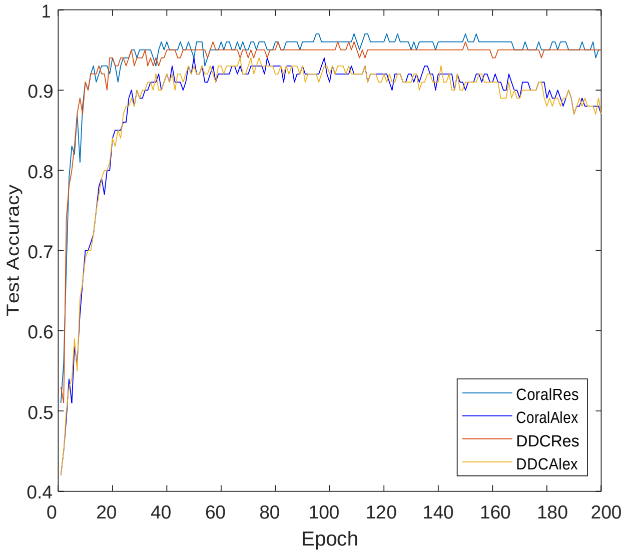


Figure 3.10: Accuracy Comparison.

with a dynamic loss function. As we can tell from Figure3.11, dynamic loss function does not only faster convergence but also gives a smoother curve. More importantly, the concept of 4D loss can be generalized to more different distribution measurements by using a dynamical combination.

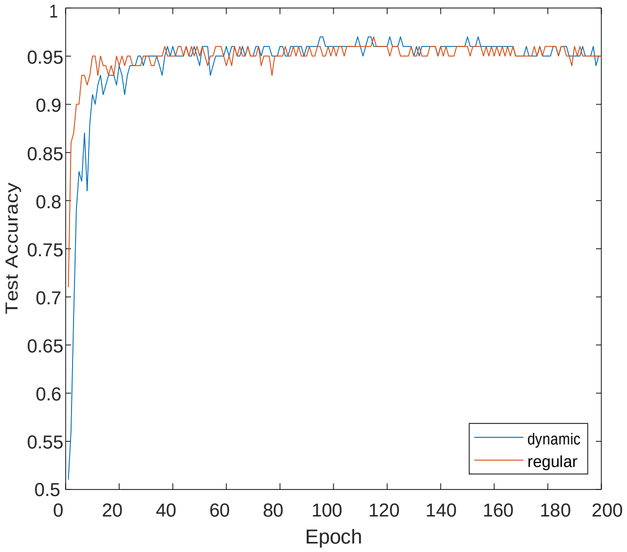


Figure 3.11: Dynamic loss vs Regular loss.

3.3.6 Concluding Remarks

First of all, recycling is an essential process for our Earth. Pollution has caused a number of species extinctions, and the number is still increasing.

Secondly, DL is one of the most powerful ways for many computer vision tasks. However, most DL methods have heavily relied on the Big Data and computational power to output state-of-art performances. In other words, the Big Data is not only the power of DL, but also the limitation of it. To address this issue, transfer learning has attracted more and more attention in the past few years, and many TL algorithms have been proven to be successful. As introduced by Andrew Ng at NIPS 2016, TL will become the main direction of DL in the future.

Finally, in this waste sorting experiment, the author first justified that TL models achieved the best performance better than all existing models built on TrashNet. And then, the novel domain loss function $4D$ proposed by us has shown the potential to benefit the TL models significantly with more accurate domain loss measurement. As in the future, few ideas can potentially push the results to an even higher level. First, GANs-based data augmentation might perform better than traditional data augmentation techniques. Then, other metrics that can calculate the distance between two different domains could also enhance the performance. Lastly, models built in this experiment used labeled-target data for training. However, other TL methods do not require labeled-target for training, which might be more helpful for those real-world problems that do not have adequate labeled data.

Chapter 4

Feature-based Distant Domain Transfer Learning with Application on Medical Imaging

In this chapter, the author studies a not well-investigated but important transfer learning problem termed Distant Domain Transfer Learning (DDTL). This topic is closely related to negative transfer. Unlike conventional transfer learning problems which assume that the source domain and the target domain are more or less similar to each other, DDTL aims to make efficient transfers even when the domains or the tasks are completely different. As an extreme example in image classification, there are only a sufficient amount of unlabeled images of watches, airplanes, and horses in the source domain, and the target domain only has a small set of labeled human face images. Previously, a few instance-based distant domain transfer algorithms were proposed to deal with this type of binary distant domain image classification problems. However, most existing algorithms are very task-specific and they are only good at binary classification tasks. In this study, the author proposes a novel feature-based distant domain transfer learning algorithm, which requires only a tiny set of labeled target data and unlabeled source data from completely different domains. Instead of selecting intermediate instances, the author introduces Distant Feature Fusion (DFF), a novel feature selection method, to discover general features cross distant domains and tasks by using convolutional autoencoder with a domain distance measurement as a feature extractor. As the novelty of this study, it can effectively handle both distant domain mutil-class image classification and binary image classification problems. More importantly, it has achieved up to 19% higher classification accuracy than ”non-transfer” algorithms, and up to 9% higher than existing distant transfer algorithms.

Moreover, In this study, the author applies the DDTL model to COVID-19 diagnose using unlabeled Office-31, Caltech-256, and chest X-ray image data sets as the source data, and a small set of labeled COVID-19 lung CT as the target data. The main contributions of this study are: 1) the proposed method benefits from unlabeled data in distant domains which can be easily accessed, 2) it can effectively handle the distribution shift between the training data and the testing data, 3) it has achieved 96% classification accuracy, which is 13% higher classification accuracy than ”non-transfer” algorithms, and 8% higher than existing transfer and distant transfer algorithms.

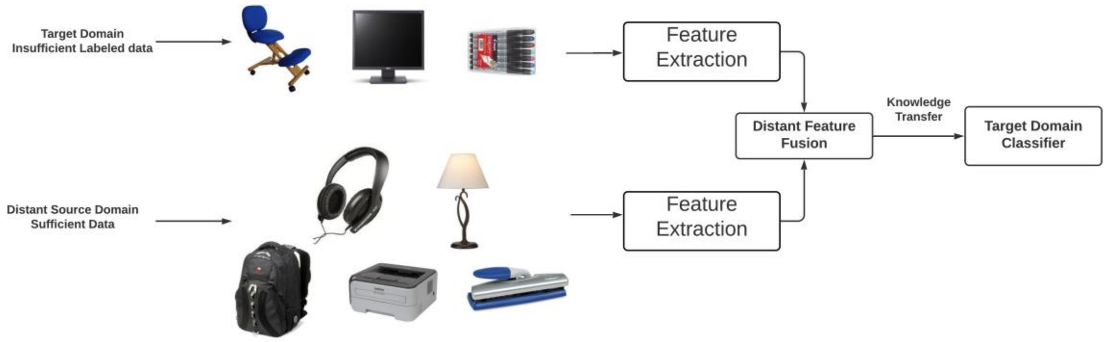


Figure 4.1: Distant Domain Transfer Learning

4.1 Feature-based Distant Domain Transfer Learning

4.1.1 Introduction

Machine learning (ML) has enabled a wide variety of beneficial applications and services [152, 154, 155, 157, 158, 171–174]. Transfer learning has the potential to improve ML in the target task by leveraging knowledge from the source task [175].

It has been proved that transfer learning is able to handle two critical machine learning problems: 1) insufficient training data, and 2) domain distribution mismatch. Theoretically, transfer learning algorithms aim to develop robust target models by using only a small set of target training data and transferring knowledge learned from other domains and tasks. Previously, the concept of adaptation layer with domain distance measurements was first proposed by [17]. It allows us to transfer knowledge between deep neural networks. In general, conventional transfer learning algorithms assume that the source domains and the targets share a certain amount of common information. However, this assumption does not always hold in many real-world applications, such as medical image processing [18, 19], rare species detection [20] and recommendation systems [21, 22]. In addition, transferring between two loosely related domains usually causes negative transfer [23–25], meaning that the knowledge transfer starts hurting the performance on the task in the target domain, and produces worse performance than non-transfer models. For instance, building a dog classification model by directly transferring knowledge from a car classification model is likely to lead to negative transfer due to the weak connection between the two domains. Therefore, it is not always feasible to apply transfer learning to areas where we cannot easily obtain enough source domain data related to the target domain.

Previously, a novel algorithm, [77] first introduced a fairly new transfer learning method, Distant Domain Transfer Learning (DDTL). As shown in Figure 4.1, DDTL aims to address the issue of negative transfer caused by loose relations of the source domains and the target domains. In other words, it allows us to safely and effectively perform the knowledge transfer when the source domains and the target domains only share a very weak connection. The inspiration behind DDTL is that the ability of a human being to learn a new thing by using knowledge learned from a number of seemingly independent things. For example, a human who knows birds and airplanes can recognize a rocket even without seeing any rockets previously. Therefore, DDTL greatly extends the use of transfer learning to more areas and applications there do not always have adequate related source data. However, this is one of the most challenging problems in transfer learning, and there are not many studies in this area.

There are few proposed distant transfer algorithms [77, 79], but most of them are task-specific and lack stability in performance. In this study, as inspired by an instance-based method [79] and multi-task learning [176], the author proposes a novel feature-based DDTL algorithm to solve image classification tasks. There are two main improvements made by our algorithm. First, the proposed algorithm does not require any labeled source domain data, and the domain can be completely different from the target domain. It only needs a tiny amount of labeled target domain to produce very promising classification accuracy on the target domain. Second, it only focuses on the target task in the target domain. To the best of our knowledge, it is the first time that distant feature extraction has been introduced in distant transfer learning. the author proposes a novel feature selection method, Distant Feature Fusion (DFF), to discover general features across distant domains and tasks by using convolutional autoencoder with a domain distance measurement. the author shows that the proposed DFF algorithm has achieved the highest accuracy on an image classification task, which has a small set of labeled target data and some unlabeled source data from different domains. Compared with transfer learning methods, supervised learning methods, and existing distant domain transfer learning methods, DDF has up to 18% classification accuracy.

The remainder of this chapter is structured as follows: In Section 4.1.2, the author first reviews the most recent DTTL works. And then, the author formulates the problem definition in Section 4.1.3. And then, the author presents the details of the proposed algorithms in Section 4.1.4. After that, the author demonstrates experimental results and analysis in Section 4.1.5. Lastly, the author concludes the chapter and discuss future directions in Section 4.1.6.

4.1.2 Related Work

Recently, insufficient training data and domain distribution mismatch have become the two most difficult challenges in the machine learning area. To address these two issues, transfer learning has emerged more and more attention due to its training efficiency and domain shift robustness. However, transfer learning also suffers from a critical issue, negative transfer [75], which significantly limits the use and performance of transfer learning. In this section, the author introduces some related works in three fields: conventional transfer learning, DDTL, and multi-task learning.

First of all, transfer learning aims to find and transfer the common knowledge in the source domain and the target domain. Furthermore, [46] expanded the use of transfer learning from traditional machine learning models to deep neural networks. Typically, there are two types of accessible transfer learning: feature-based and instance-based. And both types focus on closing the distribution distance between the source domain and the target domain. In instance-based algorithms, the goal is to discover source instances that are similar to target instances, so that the highly unrelated source samples would be eliminated. Differently, feature-based algorithms aim to map source features and target features into a common feature space where the distribution mismatch is minimized. However, both of them naturally assume that the source domain and the target domain share a fairly strong connection. Unlike conventional transfer learning, our work can transfer knowledge between different domains and tasks that are not closely related.

Secondly, most DDTL algorithms are similar to multi-task learning [118], which also benefits from shared knowledge in multiple different but related domains. Generally, multi-task learning tends to improve the performance on all the tasks. Differently, DDTL only focuses on using the knowledge in other domains to improve the performance on the target task in the target domain.

Lastly, most previous studies of DDTL focus on instance-based methods and tend to take advantage of massive related source data. Firstly, there were a few proposed instance-based DDTL algorithms [77, 79, 177] previously. For instance, the first study in this field was [77], transitive transfer learning (TTL). It transfers knowledge between text data in the source domain and the image data in the target domain by using annotate image data as a bridge. However, this algorithm is highly case-dependent and unstable on performance. At a later time, [79] introduced another instance-based algorithm with a novel instance selection method, Selective Learning Algorithm (SLA). Moreover, it uses SLA to select helpful instances from a number of unrelated intermediate domains to expand the volume of the source domain. However, this algorithm was proposed to handle binary classification problems. Furthermore, [78] proposed another feature-based method to deal with scarce satellite image data. It predicts the poverty based on the daytime satellite image by transferring knowledge learned from an object classification tasks with the help of some nighttime light intensity information as a bridge. However, this method heavily relies on a massive amount of labeled intermediate training data, which can be too expensive to apply. Different from existing DDTL algorithms, our method benefits from multiple source domains without labeled data, and those source domains can have significant discrepancies. And our method can also handle multi-class classification and consistently produce promising results.

4.1.3 Problem Statement

In this DDTL problem, the author assumes that the data of each target domain is not enough to train a robust model. And there are a number of unlabeled source domains denoted as:

$$S = \{ (x_1^1, \ldots, x_1^n), \ldots, (x_{S_N}^1, \ldots, x_{S_N}^n) \quad , \tag{4.1}$$

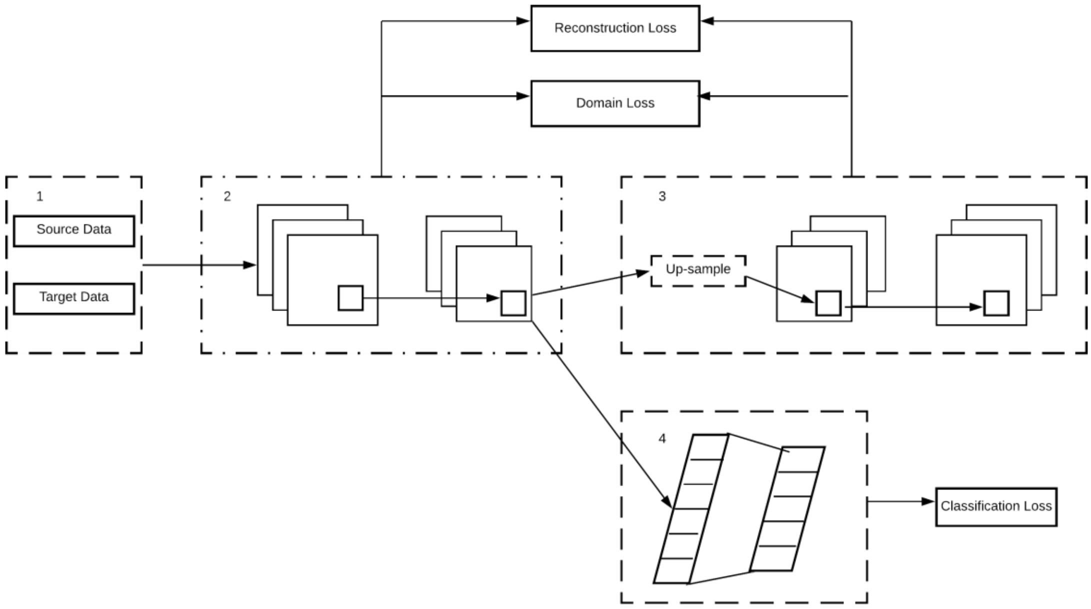


Figure 4.2: DFF Architecture

where n and S_N represent the number of samples in each source domain and the number of source domains. And then there is one or multiple labeled target domains denoted as $T = \{ [(x_1^1, y_1^1), ..., (x_1^n, y_1^n)], ..., [(x^1_{T_N}, y^1_{T_N}), ..., (x^n_{T_N}, y^n_{T_N})]$, where n and T_N represent the number of samples in each target domain and the number of target domains. Let $P(x)$, $P(y|x)$ be the marginal and the conditional distributions of a data set. In this DDTL problem, we have

$$P_{S_1}(x) = P_{S_2}(x) = ... = P_{S_N}(x) \quad P_{T_1}(x) = P_{T_2}(x) \quad ... \ = P_{T_N}(x), \tag{4.2}$$

$$P_{T_1}(y|x) \ = P_{T_2}(y|x) \ = ... = P_{T_N}(y|x). \tag{4.3}$$

The proposed work's main purpose is to develop a model for the target domain with a minimal amount of labeled data by finding generic features from distant unlabeled source domain data. The motivation behind this study is that data in distant domains is usually seemingly unrelated in instance-level but related on the feature-level. However, the connection on the feature level from one distant domain can be too weak to be used to train an accurate model. As such, simply using one or two sets of source data is likely to fail on building the target model. Therefore, the author leverages from multiple unlabeled distant source domains to obtain enough information for the target task.

4.1.4 Methodology

In this section, the author introduces a novel feature-based DDTL algorithm, Distant Feature Fusion. As shown in Figure 4.2, there are three main components in DFF: distant feature extractor, distant feature adaptation, and the target classification. There are three types of losses from three components: reconstruction loss, domain loss, and classification loss.

4.1.4.1 Distant Feature Extraction

As one of the inspirations of this study, a convolutional autoencoder pair is used as a feature extractor in DFF. As a variant of autoencoders, convolutional autoencoders [178] are usually beneficial to unsupervised image processing related problems. First of all, a convolutional autoencoder is a feed-forward neural network working in an unsupervised manner, which suits this DDTL problem perfectly since there is no labeled data in source domains. Generally, a convolutional autoencoder pair contains one input layer, one output layer, one up-sampling layer, and multiple convolutional layers. Moreover, there are two main components: encoder $E_{Conv}(\cdot)$ and decoder $D_{Conv}(\cdot)$.

Moreover, the standard process of convolutional autoencoder pairs can be demonstrated as:

$$Encoding : \boldsymbol{f} = E_{Conv}(x), Decoding : \hat{x} = D_{Conv}(\hat{f}), \tag{4.4}$$

where $\boldsymbol{f}$ is the extracted features of x, and $\hat{x}$ is the reconstructed x. Furthermore, the way to tune the parameters of a convolutional autoencoder pair is to minimize the reconstruction error on all the training instances. Conceptually, the output of the encoder can be considered as high-level features of the unlabeled training data. Furthermore, these features are learned in an unsupervised manner, so they are robust if the reconstruction error is lower than a certain threshold.

Algorithm 2: Distant Feature Fusion Algorithm

```
Input: $S = X_S$, $T = X_T$ , $Y_T$ .
        Max Iteration: I, Batch Number: N.
for $i = 1, ...., I$ do
    for $j = 1, ...., N$ do
        Feature Extraction: $f_S = E_{Conv}(X_S)$, $f_T = E_{Conv}(X_T)$ Instance
         Reconstruction: $\hat{X_S} = D_{Conv}(X_S)$, $\hat{X_T} = D_{Conv}(X_S)$
         Label Prediction: $X^T_{Pred} = C_T(f_T)$
         Calculate $L_R$, $L_D$, $L_C$
         Update $\theta_E$, $\theta_D$, $\Theta_C$
    end
end
Output: $X^T_{Pred}$
```

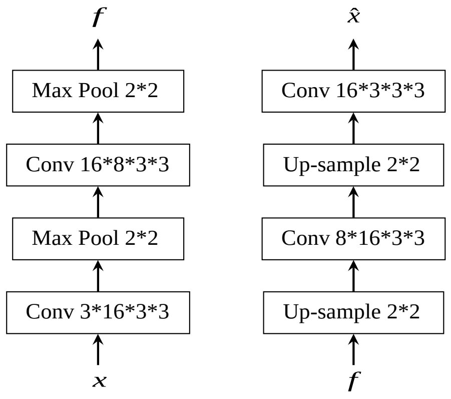


Figure 4.3: Encoder and Decoder

In this DDTL problem, as shown in Figure 4.2, the unlabeled data from all source domains are assigned with the same artificial label, 0. Differently, all the target data keep their labels. And then, the author uses a pair convolutional autoencoder to discover robust feature representation from unlabeled source domain data sets and the labeled target data sets simultaneously. And more, $Module2$ and $Module3$ are the encode and the decoder. Moreover, the structures of the encoder and the decoder can be found in Figure 4.3. There are two convolutional layers and two pooling layers in each of the encoder and the decoder. And up-sampling is applied to the encoder to ensure the quality of the reconstructed images. The process of feature extraction has three main steps: feature extraction, instance reconstruction, reconstruction measurement. First, the author feeds both the source data and the target data into the encoder to obtain high-level features f_S and f_T. And then, extracted features are sent into decoder to get reconstructions, $\hat{\boldsymbol{f}}_S$ and $\hat{\boldsymbol{f}}_T$. The equations of the first two steps are expressed as:

$$f_S = E_{Conv}(X_S), f_T = E_{Conv}(X_T); \tag{4.5}$$

$$\hat{X}_S = D_{Conv}(\boldsymbol{f}_S), \hat{X}_T = D_{Conv}(\boldsymbol{f}_T); \tag{4.6}$$

Finally, the author defines the reconstruction errors from both the source domains and the target domains as the loss function of the feature extractor, L_R is defined as:

$$L_R = \sum_{i=1}^{S_N}\sum_{j=1}^{n_{S_i}} \frac{1}{n_{S_i}}(\hat{X}^{j}_{X_{S_i}} - X^{j}_{X_{S_i}})^2 + \sum_{i=1}^{S_T}\sum_{j=1}^{n_{T_i}} \frac{1}{n_{T_i}}(\hat{X}^{j}_{X_{T_i}} - X^{j}_{X_{T_i}})^2. \tag{4.7}$$

4.1.4.2 Distant Feature Adaptation

Commonly, minimizing the reconstruction error L_R can discover a set of high-level features of the given input data. However, the distribution mismatch between the source and the target domains is significant, so minimizingL_R alone is not enough to extract robust and domain-invariant features. Therefore, the author needs extra side information to close the domain distance, so the extracted features can be robust to both the source domains and the target domains. In this research, as shown in Figure 4.2, the author adds a distant feature adaptation layer to the convolutional autoencoder pair to measure the domain loss, L_D. The maximum mean discrepancy (MMD) [179], an important statistical domain distance estimator, is used as the domain distance measurement metric. The domain loss is expressed as:

$$L_D = MMD\left(\sum_{i=1}^{S_N}\sum_{j=1}^{n_{S_i}} f^{j}_{S_i}, \sum_{i=1}^{S_T}\sum_{j=1}^{n_{T_i}} f^{j}_{T_i}\right), \tag{4.8}$$

$$MMD(X,Y) = \left\| \frac{1}{n_1}\sum_{i=1}^{n_1}\phi(x_i) + \frac{1}{n_2}\sum_{j=1}^{n_2}\phi(y_j) \right\|, \tag{4.9}$$

where n_1 and n_2 are the numbers of instances of two different domains, and $\phi(\cdot)$ is the kernel that converts two sets of features to a common reproducing kernel Hilbert space (RKHS) where the distance of two domains is maximized.

Table 4.1: Accuracy (%) of Experiments on Caltech-256

	CNN	SVM	ASVM	DTL	TTL	SLA	DFF
Target-Face	83 ± 1	84 ± 2	76 ± 4	88 ± 2	78 ± 2	96 ± 2	**98 ±1**
Target-Watch	77 ± 2	75 ± 5	60 ± 5	68 ± 3	67 ± 4	88 ± 4	**97 ± 1**
Target-Gorilla	80 ± 1	75 ± 1	54 ± 2	62 ± 3	65 ± 2	84 ± 2	**91 ± 1**

Table 4.2: Accuracy (%) of Experiments on Office-31 (Conventional Methods)

Conventional Methods	CNN	SVM	ASVM	DTL
Target-Chair	85 ± 3	83 ± 1	74 ± 2	91 ± 3
Target-Chair Monitor	79 ± 2	80 ± 2	76 ± 2	84 ± 1
Target-Chair Monitor Pen	74 ± 2	76 ± 3	62 ± 2	78 ± 2

Table 4.3: Accuracy (%) of Experiments on Office-31 (DDTL Methods)

DDTL Methods	DFF			SLA		
Target-Chair	$94 \pm 1(A-W)$	$93 \pm 2(W-A)$	$95 \pm 1(D-W)$	$92 \pm 2(A-W)$	$87 \pm 1(W-A)$	$90 \pm 3(D-W)$
Target-Chair Monitor	$91 \pm 1(A-W)$	$93 \pm 2(W-A)$	$96 \pm 2(D-W)$	$84 \pm 2(A-W)$	$82 \pm 1(W-A)$	$86 \pm 1(D-W)$
Target-Chair Monitor Pen	$85 \pm 2(A-W)$	$89 \pm 1(W-A)$	$91 \pm 1(D-W)$	$78 \pm 3(A-W)$	$72 \pm 1(W-A)$	$80 \pm 4(D-W)$

4.1.4.3 Target Classifier

Furthermore, with extracted high-level features, the author adds two fully-connected layers after the encoder to build a target classifier, C_T, for the target task in the target domain. As the motivation of this step, [10] proves that convolutional layers can discover features, and fully-connected layers can find the best feature combination for each class in the target task. In other words, fully-connected layers do not learn more new features but connect each class to a specific set of features with different weights. In this work, there is only one fully-connected layer followed by the output layer with cross-entropy loss, L_C:

$$L_C = -x[Class] + \sum_{i=1}^{T_N} \sum_{j=1}^{n_{T_i}} exp(X_T^j). \tag{4.10}$$

Finally, by embedding all three losses from 4.7, 4.8, and 4.10, the overall objective function of DFF is formulated as:

$$\underset{\theta_E,\theta_D,\Theta_C}{\text{Minimize}} \quad L = L_R + L_D + L_C, \tag{4.11}$$

where $\theta_E, \theta_D, \Theta_C$ are the parameters of the encoder, decode, and the classifier, respectively. Moreover, L is the final loss constructed by the reconstruction error, domain loss, and classification loss. Finally, all the parameters are optimized by minimizing the objective function in Equation 4.11.

4.1.4.4 Algorithm Summary

Lastly, an overview of the proposed work is summarized in Algorithm 2.

4.1.5 Experiment and Analysis

In this section, the author introduces a number of benchmark models, such as supervised learning models, conventional transfer learning models, and DDTL models. Then the author demonstrates six experiment setups. Finally, the author represents results from the proposed DFF model and the comparisons with benchmark models.

4.1.5.1 Benchmark Models

Firstly, the author selects two supervised baseline models: support vector machine (SVM) [180] and convolutional neural works (CNN) [10]. For SVM, the author chooses to use linear kernels. Moreover, for CNN, the model is constructed with two convolutional layers with 3 × 3 kernels followed by a 2 × 2 max polling kernel. Secondly, the author also chooses two conventional transfer learning models: deep transfer learning (DTL) and adaptive SVM (ASVM) [181]. Lastly, the author picks two DDTL methods: transitive transfer learning (TTL) [77] and selective learning algorithm (SLA)

[79]. However, neither of the DDTL models can be completely reproduced with many details not being introduced in the papers, and no source code is provided. As such, reproduced accuracy of those two algorithms are not as high as claimed in original papers. Therefore, the author uses the best results claimed in the original papers as benchmarks.

4.1.5.2 Date Sets and Experiment Setups

Firstly, the author conducts three experiments on Caltech-256 [182], which is an image data set that includes labeled data of 256 different classes. For each class, the number of instances is from 80 to 827. To ensure the distance between different classes, the author randomly picks six distant categories: “*watch*”, “*airplane*”, “*horse*”, “*gorilla*”, “*billiards*”, “*fa* In each experiment, the author picks one of the six classes as the target domain. Specifically, “*face*”, “*watch*”, and “*gorilla*” are chosen as target domains in the three experiments. All the source instances are considered as negative samples, and the target instances are set as positive samples. Under this setting, the experiments are formed as binary image classification problems.

Furthermore, the author uses Office-31 [183] to set up more experiments to extend the DFF algorithm to multi-class image classification problems. Office-31 has three collections of total 4110 instances from three different data sources: “amazon”, “webcam”, and “dslr”. In all three experiments, the author randomly selects five classes as source domains: S = “*backpack*”, “*lamp*”, “*printer*”, “*punchers*”, “*headphones*”. However, for three experiments, there are three different target domain setups: “*chair*”, “*chair*”, “*monitor*”, and “*chair*”, “*monitor*”, “*pen*”. Furthermore, the author also performs three knowledge transfers in each experiment, namely “*amazon*”*to*“*webcam*”, “*webcam*”*to*“*dslr*”, “*dslr*”*to*“*webcam*”.

4.1.5.3 Performance and Analysis

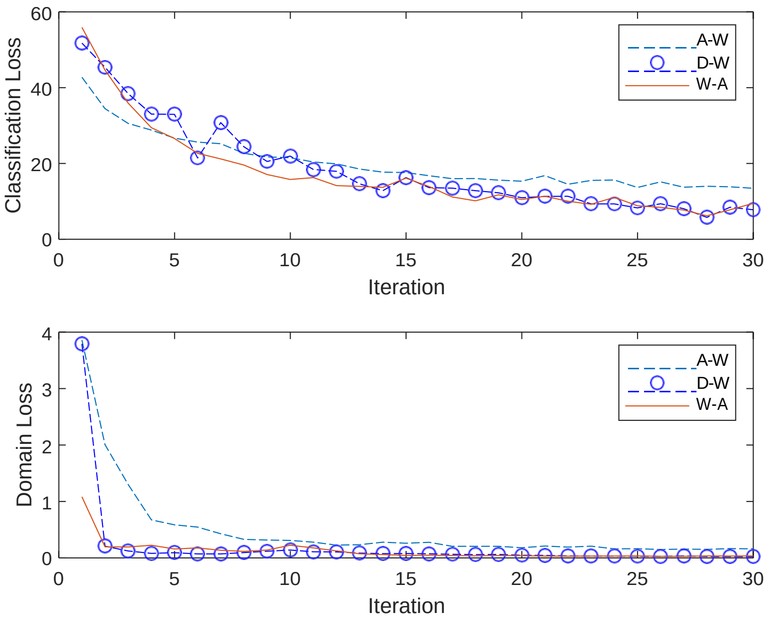


Figure 4.4: Classification Loss and Domain Loss on Office-31 Data. There are three setups: A-W (Amazon - Webcam), D-W (Dslr - Webcam), and W-A (Webcam - Amazon)

First of all, the author runs each experiment ten times to obtain each method's performance variation range. As shown in Table 4.1, with insufficient labeled training data, non-transfer methods do not carry out promising results. Conventional transfer learning algorithms carry out the worst results due to negative transfer caused by large domain discrepancies. Moreover, DTTL algorithms hold the best accuracy among three learning types of methods, and the performance of the proposed DFF algorithm has bypassed the previous record holder (SLA). It has achieved the highest accuracies in all three experimental setups. However, the first three setups are simple binary classification problems. Therefore, the author conducts a series of multi-class image classification experiments to examine the proposed DFF algorithm's performance. The accuracies of multi-classification problems on the Office-31 data set are demonstrated by Table 4.2 and Table 4.3. Additionally, the number of instances of each data source in

Office-31 varies significantly, so there are a few accuracy jumps between data sources. In Table 4.1, it shows results of all conventional methods: non-transfer models and transfer models.

Moreover, All non-transfer models are trained on the *amazon* data source, which has the most instances. As we can tell, non-transfer models' performances are relatively poor, and the accuracy drops as the number of target classes increases. Intuitively, it is caused by insufficient training data, leading the model to over-fit on the training set. Moreover, conventional transfer modes achieve better results, and the DTL model shows the best performance. Furthermore, as shown in Table 4.3, the overall performance of DDTL on multi-class classification problems is better than traditional methods. However, the classification accuracy still decreases as the increase of the number of classes. What is more, the proposed method has bypassed the performance of the previous model (SLA) in all experiments. Furthermore, the highest accuracy achieved by the DFF algorithm is 96%.

Moreover, Figure 4.4 illustrates the domain distance changing through the training and demonstrates that the final classification is closely related to the domain loss. $A - W$ has the largest domain discrepancy, which leads to the lowest classification accuracy. Furthermore, it also shows that the distant feature adaption layer can close the distribution mismatch even when domains are very distant.

4.1.5.4 Strengths and Weaknesses

The proposed DDTL algorithm, DFF, is simple and effective in dealing with image classification problems with a large discrepancy between the source and the target data sets. It solves two main challenges in deep neural networks: 1) insufficient training data and 2) significant domain mismatch. Moreover, unlike instance-based methods, DFF is a feature-based algorithm, so it does not heavily rely on a massive amount source data samples to build the bridge for knowledge. It can discover deep features that connect the source domain and the target domain with a limited amount of source data. Furthermore, it has a better generalization ability than the previous model. It is not very case-specific and domain-specific. What is more, the training process of the DFF methods is very fast and stable. Gradient explode and disappear problems do not occur like adversarial DDTL methods.

However, there are a few shortcomings of the proposed algorithm. Firstly, multi-class classification problems' performance is still not as good as conventional models trained with massive data. To address this issue, it is possible to produce cross-modality transfer, which benefits from semantic information in a domain that is in a different modality, such as from image to text. This architecture is not suitable for cross-modality transfer. Moreover, the algorithm lacks the explainability of the decision-making process required for many real-world applications. Especially for DDTL problems, interpretable methods are more helpful to us to understand and improve the model.

4.1.6 Concluding Remarks

In this chapter, the author studies the DDTL problem, there only exists a large amount of unlabeled source data and a small set of labeled target domains collected from very distant domains and tasks. Under this setting, conventional transfer learning algorithms usually suffer from the negative transfer. To address this problem, the author introduces a novel feature-based DDTL algorithm, DFF, which can effectively extract and fuse the high-level distant features learned from several distant domains. Unlike other DDTL algorithms, DFF can handle multiple source and target domains, and it does not rely on any labeled data from source domains. DFF has achieved the top performance in terms of classification accuracy compared to different types of existing algorithms. Furthermore, the author also conducts an analysis of the DFF algorithm based on different types of losses.

In the future, there are two directions regarding this DDTL problem. Firstly, the explainability of the feature-based DDTL algorithm is a challenging but essential problem. Visualizing the changes in high-level features through the training process can help us understand the domain adaptation on the feature level. Secondly, how to transfer knowledge between different fields, such as from image to audio, is also a difficult problem. Solving this problem can expand the use of transfer learning to an even further level.

4.2 Distant Domain Transfer Learning for Medical Imaging

4.2.1 Introduction

Recently, with state-of-art performance, deep learning has dominated the field of image processing [152, 184, 185]. However, deep learning methods require a massive amount of well-labeled training data, and the majority of deep leaning methods are sensitive to the domain shift [155]. Therefore, transfer learning (TL) has been introduced to deal with the issues [175, 186]. In this study, the author proposes a novel medical image classification framework. Moreover, the author implements our framework to COVID-19 diagnose with CT images. Generally, medical image data sets are difficult to access due the rarity of diseases and privacy policies. Moreover, it is not feasible to manually collect a massive amount of high-quality labeled lung CT scans associated with of COVID-19. Therefore, it is hard to develop a regular deep lea ring model with insufficient training data. To overcome this obstacle, artificial and synthetic data can be used to expand the volume of the data. However, these methods can lead to a distribution mismatch between the training data and the testing data. Furthermore, transfer learning can handle both problems simultaneously. In theory, transfer learning algorithms aim to develop robust target models by transferring knowledge from other domains and tasks. Previously, [17] proposed an adaptation layer with domain distance measurements to transfer knowledge between deep neural networks. In general, conventional transfer learning algorithms assume that the source domains and the targets share a certain amount of information. However, this assumption does not always hold in many real-world applications, such as medical image processing [18, 19], rare species detection [20] and recommendation systems [21, 22]. Moreover, transferring between two loosely related domains usually causes negative transfer [23], meaning that the knowledge transfer starts hurting the performance on the task in the target domain. For instance, building a dog classification model by directly transferring knowledge from a car classification model would likely to lead to negative transfer due to the weak connection between the two domains. Therefore, it is not always feasible to apply transfer learning to areas where we cannot easily obtain enough source domain data related to the target domain. For instance, COVID-19 diagnosis based on lung CT is a typical example where we cannot easily find related source data for training.

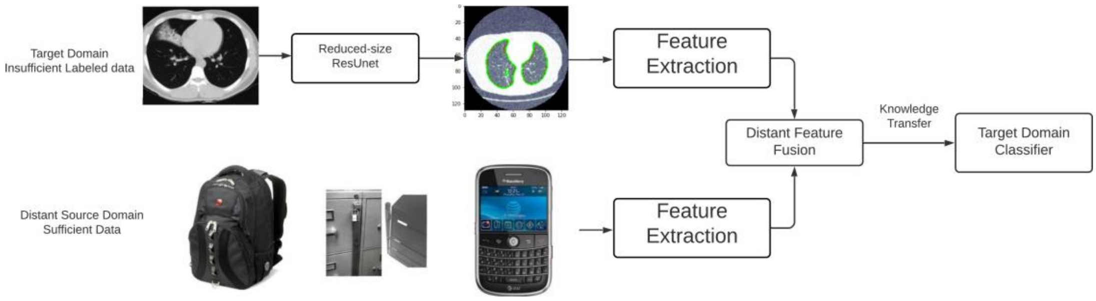


Figure 4.5: Architecture Overview of Distant Feature Fusion Model

In this study, the author develops a lung CT scan-based COVID-19 classification framework by studying a challenging problem, DDTL, which aims to deal with the shortcomings of traditional machine learning and conventional TL. As shown in Figure-4.5, the proposed framework contains two parts: semantic segmentation and DFF. It can perform knowledge transfer between seemingly unrelated domains. Moreover, DDTL [77] is a newly introduced transfer learning method that mainly aims to address the issue of negative transfer caused by loose relations of the source domains and the target domains. Unlike conventional TL methods, the proposed DDTL algorithm benefits from fusing distant features extracted from distant domains. Generally, DDTL is usually involved with situation that the source domain and the target domain have completely tasks. Moreover, the inspiration for DDTL is from the ability of human beings to learn new things by bridging knowledge acquired from several seemingly independent things. For example, a human who knows birds and airplanes can recognize a rocket even without seeing any rockets previously. Importantly, DDTL dramatically extends the use of transfer learning to more areas, and applications where do not always have adequate related source data. In this case, the author considers COVID-19 classification as a DDTL problem that can benefit from distant but more accessible domains. Furthermore, the author uses three open-source image data sets as source domain data sets to develop a robust COVID-19 classification method based on lung CT images.

Previously, there are few proposed distant transfer algorithms [77, 79], but most of them are task-specific and lack the stability in performance. Inspired by an instance-based method [79] and multi-task learning [176], the author builds a DDTL algorithm to solve COVID-19 classification tasks by extracting and fusing distant features. There are two main improvements made by our algorithm. Firstly, it does not require any labeled source domain data, and the source domains can be completely different from the target domain. The proposed model only needs a small amount of labeled target domain and can produce very promising classification accuracy on the target domain. Secondly, it only focuses on improving the performance of the target task in the target domain. To the best of our knowledge, it is the first time that DDTL has been applied to medical image classification. Furthermore, the author introduces a novel feature selection method (DFF) to discover general features across distant domains and tasks by using convolutional autoencoders with a domain distance measurement. To outline, there are four main contributions made in this study: 1) Propose a new DDTL algorithm for fast and accurate COVID-19 diagnose based on lung CT, 2) Examine existing deep learning models (transfer and non-transfer) on COVID-19 classification problem, 3) The proposed algorithms has achieved the highest accuracy on this task, which has a small set of labeled target data and some unlabeled source data from different domains. Moreover, compared with other transfer learning methods, supervised learning methods, and existing DDTL methods, the proposed DFF model has achieved up to 34% higher classification accuracy and 4) The proposed framework can be easily generalized to other medical image processing problems.

The remainder of this chapter is structured as follows: In Section 4.2.2, the author first reviews the most recent DTTL works. And then, the author formulates the problem definition in Section 4.2.3. Next, the author presents the details of the proposed algorithm in Section 4.2.4. After that, the author presents experimental results and analysis in Section 4.1.5. Lastly, the author concludes the chapter and discuss future directions in Section 4.2.6.

4.2.2 Related Work

Insufficient training data and domain distribution mismatch have become the two most challenging problems in machine learning. To address these two issues, transfer learning has emerged a lot of attention due to its training efficiency and domain shift robustness. However, transfer learning also suffers from a critical shortcoming, negative transfer [75], which significantly limits the use and performance of transfer learning. In this section, the author introduces some related works in three fields: conventional transfer learning, DDTL, and existing ML methods for COVID-19 classification.

4.2.2.1 Conventional Transfer Learning

First of all, TL methods aim to solve the target task by leveraging the common knowledge learned from source tasks in different domains, so it does not need to learn the target task from scratch with a massive amount of data. Furthermore, [46, 187, 188] expanded the use of transfer learning from traditional machine learning models to deep neural networks. Typically, there are two types of accessible transfer learning: feature-based and instance-based. Both types focus on closing the distribution distance between the source domain and the target domain. In instance-based algorithms, the goal is to discover source instances similar to target instances, so that highly unrelated source samples would be eliminated. Differently, feature-based algorithms aim to map source features and target features into a common feature space where the distribution mismatch is minimized. However, both of them assume that the source domain and the target domain share a fairly strong connection. Unlike conventional transfer learning, our work can transfer knowledge between different domains and tasks that are not closely related.

4.2.2.2 DDTL

Secondly, the setting of DDTL is similar to multi-task learning [118], which also benefits from shared knowledge in multiple close domains. Generally, multi-task learning tends to improve the performance on all tasks. Differently, DDTL only focuses on using the knowledge in other domains to improve the performance of the target task. Moreover, most previous studies of DDTL are instance-based and they tend to take the advantage of massive related source data. Firstly, [77] introduced an instance-based algorithm, transitive transfer learning (TTL). It transfers knowledge between text data in the source domain and the image data in the target domain by using annotated image data as a bridge. However, TTL is highly case-dependent and unstable in performance. Similarly, [79] introduced another instance selection method, Selective Learning Algorithm (SLA). However, this algorithm was mainly designed for binary classification problems. Differently, [78] proposed a feature-based method to deal with scarce satellite image data. It predicts the poverty based on daytime satellite images by transferring knowledge learned from an object classification tasks with the aid of nighttime light intensity information as a bridge. However, this method relies heavily on a massive amount of labeled intermediate training data. Notably, our method benefits from multiple source domains without labeled data, and those source domains can have significant discrepancies. Furthermore, our method can also handle multi-class classification while consistently producing promising results.

4.2.2.3 Machine Learning for COVID-19 Diagnosis

Moreover, to overcome the shortage of COVID-19 testing toolkits, many efforts have been made to search for alternative solutions. Several studies [189–191] introduced machine techniques to COVID-19 diagnosis, including but not limited to, convolutional neural networks (CNN), transfer learning, empirical modeling. However, most existing non-transfer models suffer from a common shortcoming that is insufficient well-labeled training data. Transfer leanings methods can carry out fairly decent classifications, but they are still limited by the domain discrepancy between the source data and the target data.

4.2.3 Problem Statement

In this DDTL problem, the author assumes that the data of each target domain is insufficient to train a robust model. And there are a number of unlabeled source domains denoted as $S = \{ (x_1^1, ..., x_1^{n_{S_1}}), ..., (x_{S_N}^1, ..., x_{S_N}^{n_{S_N}})$, where n and S_N represent the number of samples in each source domain and the number of source domains. Then the author denotes one or multiple labeled target domains as:

$$
\begin{gathered}
T = [(x_1^1, y_1^1), ..., (x_1^{n_{T_1}}, y_1^{n_{T_1}})], \\
..., [(x_{T_N}^1, y_{T_N}^1), ..., (x_{T_N}^{n_{T_N}}, y_{T_N}^{n_{T_N}})]
\end{gathered}
\tag{4.12}
$$

, where n and T_N represent the number of samples in each source domain and the number of source domains. Let $P\,(x)$, $P\,(y|x)$ be the marginal and the conditional distributions of a data set. In this DDTL problem, the author has the following:

$$P_{S_1\text{-}S_N}(x) \;=P_{T_1\text{-}T_N}, \tag{4.13}$$

$$P_{T_1}(y|x) \;=P_{T_2}(y|x) \;= ... = P_{T_N}(y|x). \tag{4.14}$$

The main objective of the proposed work is to develop a model for the target domain with a minimal amount of labeled data by finding generic features from distant unlabeled source domain data. The motivation behind this study is that data in distant domains is usually seemingly unrelated in the instance-level but related in the feature-level. However, the connection on the feature level from one distant domain can be too weak to be used to train an accurate model. As such, simply using one or two sets of source data is likely to fail in building the target model. Therefore, the author leverages from multiple unlabeled distant source domains to obtain enough information for the target task.

4.2.4 Methodology

In this section, the author introduces the proposed COVID-19 diagnose framework. Firstly, the author presents the reduced-size ResNet segmentation model. After that, the author introduces the novel DDTL algorithm, DFF.

4.2.4.1 Lung CT Segmentation by Reduced-size ResUnet

First of all, extracting features from a full size lung CT image with a small training set can be difficult because the model might end up focusing on noise in the useless parts of the images. Therefore, it is important tp pre-process the image by applying semantic segmentation. As shown in Figure-4.6, the author can remove random noise and preserve the important information in the lung area of a image. Moreover, a small data set for training can lead to a over-fitting for a deep neural network. Therefore, the author develops a reduced-size ResNet for this Covid-19 diagnose task.

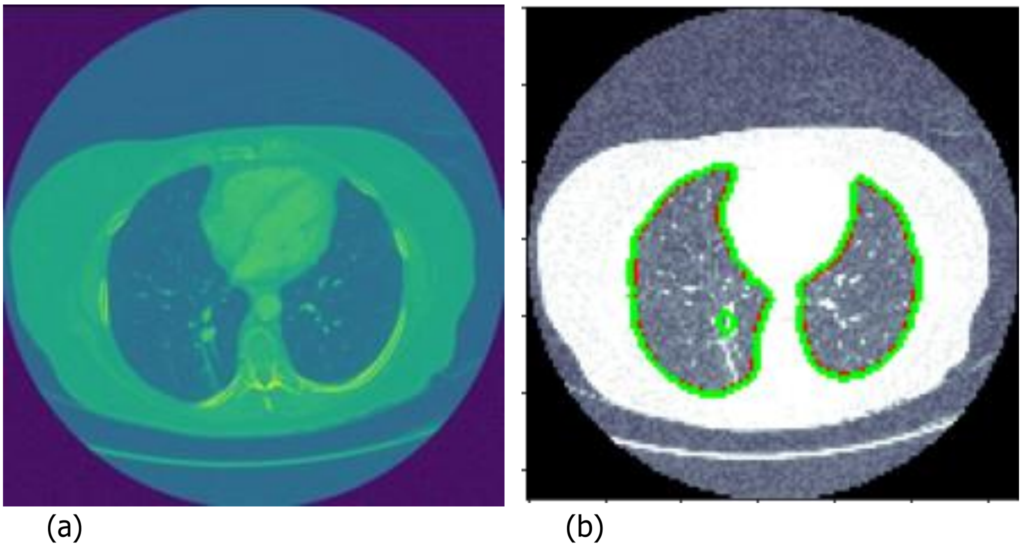


Figure 4.6: 4.6a Original Image. 4.6b Segmented Image.

Fisr of all, the proposed reduced-size ResUnet [192] contains two feature extraction parts: four convolutional blocks layers with down-sampling and four deconvolutional layers with up-sampling. Moreover, the author reduces the numbers of convolutional layers and deconvolutional layers, and apply dropout layers to prevent over-fitting. Furthermore, I adopt skip-connection to prevent two main problems in the training process: gradient explode and gradient disappear. In this study, the author implements a single skip-connection to form convolutional and deconvolutional blocks. By doing this, the convergence time of the model is faster and the training process is more stable.

Commonly, image segmentation tasks require to perform accurate pixel-level classification on the input images. Therefore, it is critical to design a proper loss function based on each task. In this study, the final loss function is composed by a soft-max function over the last feature map combined with the cross-entropy loss. The expressions of the soft-max function and cross-entropy functions are:

$$p_k(x) = exp(f_k(x))/\mathop{}_{k=1}^{K} exp(f_k(x)), \tag{4.15}$$

$$E = \mathop{}_{x} \omega(x)log(p_{(l(x))}(x)), \tag{4.16}$$

where $f_k(x)$ represents the activation map of the kth feature at xth pixel and K is the total number of classes, and the cross-entropy penalizes at each position the deviation of $p_{(l(x))}$. Furthermore, the segmentation boarder is computed with morphological operations. The weight map is expressed as:

$$\omega(x) = \omega_c(x)\omega_0 exp(-\frac{(d_1(x) + d_2(x))^2}{2\sigma^2}), \tag{4.17}$$

where ω_c is the weight map to balance the class frequencies, d_1 and d_2 are the distances between a pixel to the closest boarder and the second coolest boarder, and ω_0 and σ are the initialization values.

4.2.4.2 DFF

As shown in Figure-4.7, there are three main components in DFF: distant feature extractor, distant feature adaptation, and the target classification. There are three types of losses from three components: reconstruction loss, domain loss, and classification loss.

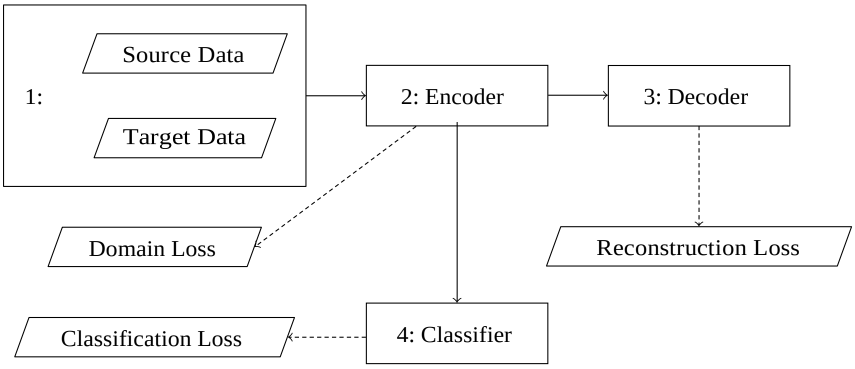


Figure 4.7: DFF Architecture: there are three main components in DFF, distant feature extractor, distant feature adaptation, and the target classification. There are three types of losses from three components: reconstruction loss, domain loss, and classification loss.

Distant Feature Extraction

As one of the inspirations of this study, a convolutional autoencoder pair is used as a feature extractor in DFF. convolutional autoencoders [178] usually benefit unsupervised image processing related problems. Firstly, a convolutional autoencoder is a feed-forward neural network working in an unsupervised manner, which suits this DDTL problem perfectly since there is no labeled data in source domains. Moreover, there are two main components: encoder $E_{Conv}(\cdot)$ and decoder $D_{Conv}(\cdot)$. The standard process of convolutional autoencoder pairs can be demonstrated as:

$$Encoding : \boldsymbol{f} = E_{Conv}(x), Decoding : \hat{x} = D_{Conv}(\hat{f}), \tag{4.18}$$

where $\boldsymbol{f}$ is the extracted features of x, and $\hat{x}$ is the reconstructed x. In addition, the way to tune the parameters of a convolutional autoencoder pair is to minimize the reconstruction error on all the training instances. Conceptually, the output of the encoder can be considered as high-level features of the unlabeled training data. Furthermore, these features are learned in an unsupervised manner, so they are robust if the reconstruction error is lower than a certain threshold.

In this DDTL problem, as shown in Figure-4.7, the author uses a convolutional autoencoder pair to discover robust feature representation from unlabeled source domain data sets and the labeled target data sets simultaneously. The structure of the autoencoder pair contains two convolutional layers and two pooling layers in both the encoder and decoder. Up-sampling is applied to the encoder to ensure the quality of the reconstructed images. The process of feature selection has three main steps: feature extraction, instance reconstruction, and reconstruction measurement. First, the author feeds both the source data and the target data into the encoder to obtain high-level features f_S and f_T . Then, extracted features are sent into the decoder to get reconstructions, $\hat{X}_S$ and $\hat{X}_T$. The equations of the first two steps are expressed as:

$$f_S = E_{Conv}(X_S), f_T = E_{Conv}(X_T); \tag{4.19}$$

$$\hat{X}_S = D_{Conv}(\boldsymbol{f}_S), \hat{X}_T = D_{Conv}(\boldsymbol{f}_T); \tag{4.20}$$

where X_S and X_T are the source and the target samples, and f_S and f_T are the source and the target features. Finally, the author defines the reconstruction errors from both the source domains and the target domains as the loss function of the feature extractor, L_R as follow:

$$L_R = \mathop{}\limits_{i=1}^{S_N}\mathop{}\limits_{j=1}^{n_{S_i}} \frac{1}{n_{S_i}}(X^{\hat{j}}_{X_{S_i}} - X^{j}_{X_{S_i}})^2 + \mathop{}\limits_{i=1}^{S_T}\mathop{}\limits_{j=1}^{n_{T_i}} \frac{1}{n_{T_i}}(X^{\hat{j}}_{X_{T_i}} - X^{j}_{X_{T_i}})^2. \qquad (4.21)$$

where S_N and S_T are the numbers of the source domains and the target domains, n_{S_i} and n_{S_i} are the numbers of instances in the *ith* source domain and the target domain.

Distant Feature Adaptation

Algorithm 3: Distant Feature Fusion Algorithm

```
Input: $S = X_S$, $T = X_T$ , $Y_T$ .
        Max Iteration: I, Batch Number: N.
for $i = 1, ...., I$ do
    for $j = 1, ...., N$ do
        Feature Extraction: $f_S = E_{Conv}(X_S)$, $f_T = E_{Conv}(X_T)$
         Instance Reconstruction: $\hat{X}_S = D_{Conv}(X_S)$, $\hat{X}_T = D_{Conv}(X_S)$
         Label Prediction: $X^T_{Pred} = C_T(f_T)$
         Calculate $L_R$, $L_D$, $L_C$
         Update $\theta_E$, $\theta_D$, $\Theta_C$
    end
end
Output: $X^T_{Pred}$
```

Commonly, minimizing the reconstruction error L_R can discover a certain amount of features with the given input. However, there is a large distribution mismatch between the source and the target domains, so minimizingL_R alone cannot extract enough robust and domain-invariant features. Therefore, the author needs extra side information to close the domain distance. In this research, as shown in Figure-4.7, the author adds a distant feature adaptation layer to the convolutional autoencoder pair to close the domain distance L_D. The maximum mean discrepancy (MMD) [179] is important statistical domain distance estimator. The domain loss is expressed as:

$$L_D = MMD(\sum_{i=1}^{S_N}\sum_{j=1}^{n_{S_i}} f^j_{S_i}, \sum_{i=1}^{S_T}\sum_{j=1}^{n_{T_i}} f^j_{T_i}), \tag{4.22}$$

$$MMD(X,Y) = \left\| \frac{1}{n_1}\sum_{i=1}^{n_1}\phi(x_i) + \frac{1}{n_2}\sum_{j=1}^{n_2}\phi(y_j) \right\|, \tag{4.23}$$

where n_1 and n_2 are the numbers of instances of two different domains, and $\phi(\cdot)$ is the kernel that converts two sets of features to a common reproducing kernel Hilbert space (RKHS) where the distance of two domains is maximized.

Target Classifier

Furthermore, with extracted distant features, the author adds a target classifier C_T after the encoder. As the motivation of this step, [10] proves that fully-connected layers aim find the best feature combination for each class in the target task. In other words, fully-connected layers do not learn more new features but connect each class to a specific set of features with different weights. In this work, there is only one fully-connected layer followed by the output layer with cross-entropy loss, L_C:

$$L_C = -x[Class] + \sum_{i=1}^{T_N}\sum_{j=1}^{n_{T_i}} exp(X^j_{T_i}). \tag{4.24}$$

where $X^j_{T_i}$ is the *jth* sample in the *ith* target domain. Finally, by embedding all three losses from 4.21, 4.22, and 4.24, the overall objective function of DFF is formulated as:

$$\underset{\theta_E,\theta_D,\Theta_C}{\text{Minimize}} \quad L = L_R + L_D + L_C, \tag{4.25}$$

Table 4.4: Model Comparison

	CNN	Alexnet	Resnet	SelfTran	SLA	DFF
Transferable	No	Yes	Yes	Yes	Yes	Yes
Base Model	Discriminative	Discriminative	Discriminative	Discriminative	Discriminative	Discriminative
Loss Type	Entropy	Entropy	Entropy	Entropy	Entropy&MMD	Entropy&MMD
Learning Type	Feature-based	Feature-based	Feature-based	Feature-based	Instance-based	Feature-based

Table 4.5: Data Sets

Data Set	Total Classes	Total Samples	Label	Mask
Caltech-256	256	30670	Yes	No
Office-31	31	4110	Yes	No
Chest Xray	4	562	Yes	No
Lung-CT	4	367	Yes	Yes
Covid19-CT	2	565	Yes	No

where θ_E, θ_D, Θ_C are the parameters of the encoder, decoder, and the classifier, respectively. Moreover, L is the final loss constructed by the reconstruction error, domain loss, and classification loss. Finally, all the parameters are optimized by minimizing the objective function in Equation 4.25.

4.2.4.3 Algorithm Summary

Lastly, an overview of the proposed work is summarized in Algorithm 3.

4.2.5 Experiment and Analysis

In this section, the author introduces a number of benchmark models, such as supervised learning models, conventional transfer learning models, and DDTL models. Then the author sets up a serious of experiments. After that, the author demonstrates the experimental results. Finally, the author presents training details and the analysis of experimental results.

Table 4.6: Segmentation Performance

	IoU	Dice	Accuracy
Reduced-ResUnet	0.96	0.97	0.96
Unet	0.86	0.88	0.87

4.2.5.1 Benchmark Models

In this study, as shown in Table. 4.4, the author chooses several transfer models and non-transfer models for comparisons. By comparing results from different methods, the author can justify the improvements made by the proposed methods. Firstly, the author selects three supervised non-transfer baseline models: convolutional neural works (CNN), Alexnet [10], and Resnet [193]. For CNN, the model is constructed with three convolutional layers with 3 × 3 kernels followed by a 2 × 2 max pooling kernel. Secondly, the author also chooses three conventional transfer learning models: fine-tuned Alexnet, fine-tuned Resnet, and self-transfer (SelfTran) model [189]. What is more, the author chooses one instance-based DDTL method: selective learning algorithm (SLA) [79]. Furthermore, all details of each benchmark model are specified in Table. 4.4.

4.2.5.2 Date Sets and Experiment Setups

In this study, as shown in Table. 4.5, the author totally uses six open-source data sets: Caltech-256 [182], Office-31 [183], chest X-Ray for pneumonia detection [194], Lung CT [195], and Covid19-CT [196]. The first, Caltech-256 includes labeled data of 256 different classes. For each class, the number of instances is from 80 to 827. Then, Office-31 has 31 different common office objects, with total 4110 instances collected from three different data sources: ”amazon”, ”webcam”, and ”dslr”. However, Office-31 is an unbalanced data set. Moreover, the chest X-Ray data set contains 5226 well-labeled images. Intuitively, the chest X-Ray images should have the most similarity with lung X-Ray images, so the author wonders if directly transfer and fine-tune would carry out better performance than the proposed method. Moreover, Covid19-CT contains 565 labeled lung CT images: 349 positive samples, and 216 negative samples. It is considered as a fairly small data set for training deep learning models. Finally,

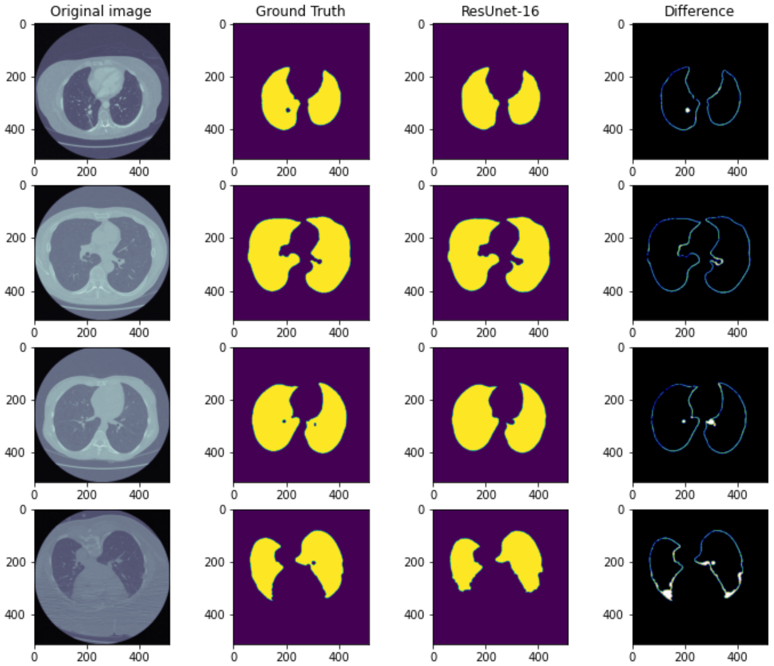


Figure 4.8: Lung CT Segmentation

Table 4.7: Top Accuracies (%) of Examined Models

	CNN	Alexnet	Resnet	SelfTran	SLA	DFF
Testing Accuracy (Raw-Image)	74 ± 1	82 ± 3	86 ± 3	83 ± 1	54 ± 2	**93 ± 1**
Testing Accuracy (Segmented-Image)	78 ± 2	85 ± 2	88 ± 3	87 ± 3	62 ± 1	**96 ± 1**

Table 4.8: Accuracies (%) of DDTL Models with Single Source Domain

Source Domain	Caltech256	Amazon	Webcam	Dslr	Chest X-Ray
SLA (Raw-Image)	54 ±2	52 ±1	48 ±2	48 ±3	52 ±4
SLA (Segmented-Image)	62 ±1	54 ±1	46 ±3	56 ±1	61 ±2
DFF (Raw-Image)	**88 ±2**	78 ±3	73 ±2	70 ±1	63 ±3
DFF (Segmented-Image)	**90 ±1**	76 ±1	76 ±2	74 ±3	69 ±2
Conventional TL Models					
Fine-tuned Alexnet (Raw-Image)	77 ± 1	61 ± 2	64 ± 1	51 ± 2	73 ± 3
Fine-tuned Alexnet (Segmented-Image)	80 ±2	64 ±1	68 ±3	52 ± 2	81 ± 1
Fine-tuned Resnet (Raw-Image)	66 ±2	57 ±3	61 ±1	54 ± 1	64 ± 2
Fine-tuned Resnet (Segmented-Image)	72 ±1	61 ±1	64 ±2	62 ± 3	65 ± 2

the author uses the lung CT data set for the segmentation model. The data set has 367 lung CT images with pixel-level masks.

Moreover, the author runs each experiment five times to investigate the performance fluctuation range. Firstly, the author produces 4 experiments on CNN and conventional TL models with the Covid19-CT data. And then, the author sets up a series of experiments on DDTL models with single source domain and multi-source domains to explore the potential of the

Table 4.9: Accuracies (%) of DDTL Models with Multiple Source Domains

Primary Source Domain	Caltech256	Amazon	Webcam	Dslr
Auxiliary Source Domain	Chest X-Ray			
SLA (Raw-Image)	54 ± 2	52 ± 1	48 ± 2	48 ± 3
SLA (Segmented-Image)	62 ± 1	55 ± 3	51 ± 1	47 ± 2
DFF (Raw-Image)	**93 ± 1**	73 ± 3	64 ± 2	86 ± 3
DFF (Segmented-Image)	**96 ± 1**	75 ± 2	66 ± 1	87 ± 1

learning method. As shown in Table. 4.8, there are five unlabeled source domains data sets: ***Caltech-256***, ***Amazon***, ***Amazon***, ***Webcam***, ***Chest X-Ray***, and one labeled target data set: ***Lung CT for Covid-19***. What is more, another regular ***Lung CT*** contains masks for segmentation. Moreover, the first four source domains are seemingly unrelated to the target domain, but the last source domain is visually related to the target domain.

Furthermore, unlike previous methods, the proposed method is able to utilize multiple source domains to improve the performance in the target domain. Therefore, as we can tell from Table. 4.9, the author chooses four primary source domains and use the ***Chest X-Ray*** data set as the auxiliary domain. In the following sections, the author will present the results and analysis.

4.2.5.3 Performance and Analysis

In this section, the author first presents the performance of the segmentation model. After that, the author gives an overview of results of all examined classification methods and present insights on performance differences. Then, the author provides training details and analysis of our proposed DDTL algorithm.

Segmentation Performance

Firstly, the most informative part of a lung CT is the lung area, and it allows machines to better imitate the behaviors of real specialists. The proposed reduced-size ResUnet is trained from scratch because there is no pre-trained model for this novel architecture. Moreover, the dropout layers and the skip-connections are applied to prevent over-fitting and non-convergence problems. As we can tell from Figure-4.6, the segmented image shows an accurate and clear contour of the lung area, so the author can select only the lung area as the input for the DFF model. Furthermore, Figure-4.8 shows a better visual results of the segmentation model. The first column presents the original image, the second column shows the ground truth of the lung area, the third column gives the pixel-level classification of the model, and the fourth column illustrates the pixel-level difference between the ground truth and the prediction.

Moreover, the author uses two common evaluation metrics for image segmentation tasks to quantify the performance. In the study, the author uses IoU (intersection over union), Dice (F1 Score), and pixel-level accuracy as the evaluation metrics. The definitions of them are:

$$IoU = \frac{TP}{TP + FP + FN}, \tag{4.26}$$

$$Dice = \frac{2TP}{2TP + FP + FN}, \tag{4.27}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}. \tag{4.28}$$

Furthermore, for the comparison, the author also conducts experiments on the original Unet with the same data set. The details are shown in Table. 4.6. Obviously, the reduced-size ResUnet outperforms the original Unet. The possible reasons are: 1) the original Unet cannot effectively prevent the model from learning noise, 2) the skip-connection helps the model to extract deeper features.

Classification Performance Overview

As demonstrated in Table. 4.7, the proposed DFF algorithm outperforms the highest test classification accuracy (96%). And more, the CNN model is only at (78%) classification accuracy. Intuitively, it is caused by insufficient training data. Moreover, the Alexnet and SelfTran output promising accuracies (85%, 88%). In theory, initializing with pre-trained parameters can boost the performance due to the pre-train data set. However, the settings are more or less similar to TL, and the accuracies are still lower than the proposed DDTL method. This performance gap can be caused by large domain discrepancy between two distant domains. The traditional models cannot close the domain distance to avoid the performance degradation. However, there is no evidence of negative transfer in the fine-tuning models. The instance-based DDTL model (SLA) has the worst accuracy (62%), which is clearly a negative transfer case. Theoretically, the instance selection by the re-weighting matrix eliminates way too many source domain samples due to a large distribution discrepancy. As such, it cannot extract sufficient information for the knowledge transfer. It can be considered as the same situation as the CNN model with insufficient training data. Furthermore, pre-processing the data with semantic segmentation can improve the performance. Moreover, it proves that preserving the most informative part by eliminating random noise from a small data set can enhance the final classification performance.

Furthermore, the author has observed other interesting things. First of all, feature-based algorithms have more promising performances on the COVID-19 classification problem. Differently, the instance-based method completely failed to solve this task. Intuitively, samples in distant domains are seemingly unrelated at the instance level, but they might still share common information at the feature level. Therefore, the instance selection method tend to miss important information with only learning features at the visual-level. Differently, the feature-based models tend to ignore the large discrepancy at the visual-level. Instead, they aim to discover the relationship of two domains at the feature-level. Therefore, it can close the distribution mismatch by extracting domain-confusing features.

Moreover, Table. 4.8 shows performances of conventional TF models and DDTL models with single source domain. Firstly, the proposed DDTL algorithm achieves the highest classification accuracy (90%, and SLA method shows negative transfer on all five source domains. It further approves that instance selection process might not be reliable for DDTL problems. However, the advantage of SLA is that it does not require labeled target data, while the proposed method needs labeled target data. In addition, not all source domains are suitable for distant knowledge transfer. The seemingly related domain, chest X-Ray, is actually not the

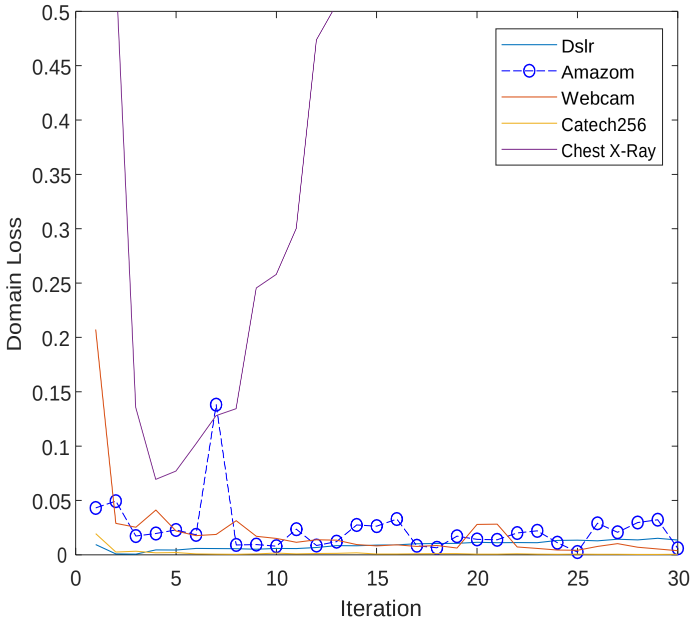


Figure 4.9: DFF Domain Losses with Single Source Domain

most transfer-friendly for this task. Other data sets that are visually distant from the target domain carry out better results. It approves the theory that seemingly unrelated domains might be statistically connected in the feature-level. The author will provide more evidences in later contents.

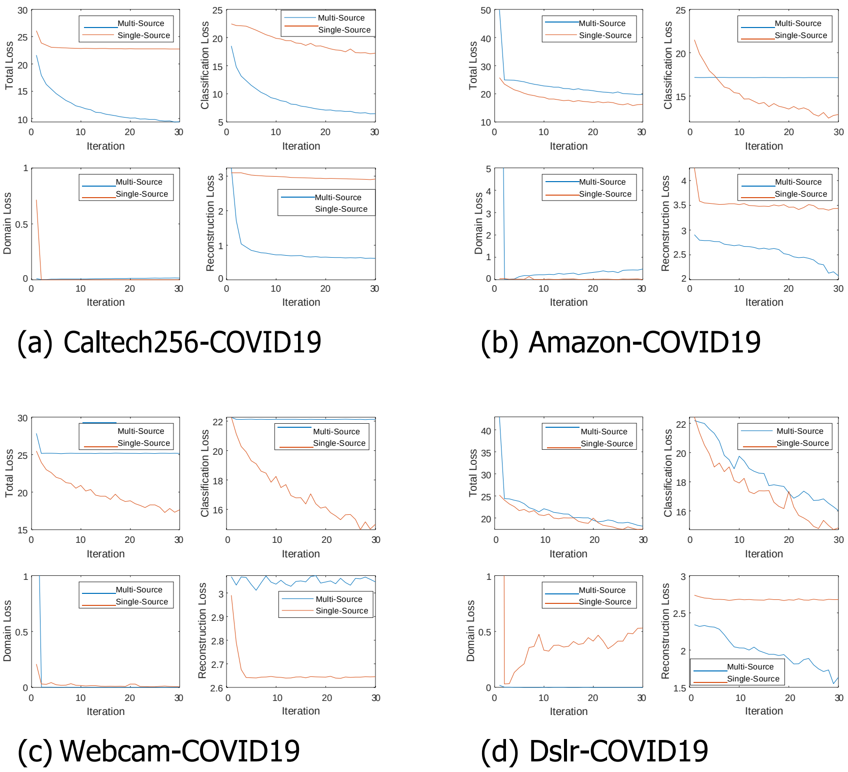
The best performance of conventional TL models is (88% which is better than non-transfer methods. Initializing with pre-trained weights only yields a faster convergence but it does not improve the performance in this case. Accuracies from experiments of ***Chest X-Ray to Covid19-CT*** turns out to be worse than other experiment setups even the chest X-Ray is commonly assumed to be the most similar to the target domain. However, as shown in Figure-4.9, the domain loss between the Covid19-Xray and chest X-Ray is the greatest in all experiments. It also proves that seemingly related domains might be distant in the feature level, so it is not always reliable to hand-pick source domains in DDTL problems.

Moreover, the enhancement from semantic segmentation is still not good enough to reach the human-level performance. Therefore, unlike most existing DDTL algorithms, the author wishes to even improve the performance by using multiple source domain. Importantly, in DDTL problems, finding shared information cross different domains is the key to perform a safe knowledge transfer. However, the amount of common information extracted from a single distant domain might not be sufficient. As shown in Table. 4.9, the proposed method achieves (96% classification accuracy with using ***Caltech-256*** as the primary source domain and ***Chest X-Ray*** as the auxiliary source domain. It means that these two data sets have less information overlapping, so the DFF model can extract more useful shared knowledge to transfer to the target domain. Differently, performance degradation appears in others multi-source domain experiments, which means others pairs have shared information that causes over-fitting.

However, one significant weakness of DDTL models is that they are highly dependent on the quantity and versatility of the source domains. As we can tell from Table. 4.8, the performances of the proposed model decreases dramatically when the webcam and the dslr data sets of Office-31 are set as the source domains. Theoretically, DDTL models benefit from extracting the common knowledge of the source domain and the target domain, but they cannot complete this type of feature extraction when the source data set is small. There are only 550 and 640 samples in the webcam and the Dslr data sets, which are less than the target samples. Therefore, it is not easy to safely and effectively transfer knowledge between different domains. On the contrary, the Caltech-31 data set has over 33000 samples from 256 different classes, so it is easier to perform the knowledge transfer.

Analysis of DFF

Figure-4.10a-4.10d shows details of the DDF models in single source domain setting and the multi-source domain settings, illustrating four types of losses: total training loss, target classification loss, domain loss, and reconstruction loss. Firstly, The proposed DFF algorithm has achieved the highest test classification accuracy when the Caltech-256 data set is the primary source domain and the chest X-Ray data set is the auxiliary source domain. Overall, it has the most smooth curves and the smallest domain loss. Moreover, with the additional information from the auxiliary source domain, its classification loss and reconstruction loss are dramatically reduced. In other words, the model is able to extract additional features from the auxiliary domain and use it as a bridge to close the distance from the target domain. Moreover,



(a) Caltech256-COVID19

(b) Amazon-COVID19

(c) Webcam-COVID19

(d) Dslr-COVID19

Figure 4.10: Training Details of experiments on ADFE with 4 setups: ***Caltech-256 to Covid19-CT*** , ***Office-31-Amazon to Covid19-CT*** , ***Office-31-Webcam to Covid19-CT***, ***Office-31-dslr to Covid19-CT***. In each sub-figure, up left is total loss, up right is target classification loss, down left is domain distance, and down right is reconstruction error.

large declines in performance appear in the other experiments with ***Amazon*** and ***Webcam***. As mentioned earlier, the performance degradation can be caused by overlapping information in the primary and the secondary source domains. The model is over-fit due to the duplicated knowledge in two source domains. Especially, in the experiment 4.10b, the domain loss is increased but the classification loss is not lowered. Furthermore, this proves that seemingly distant instances might share a certain amount of common features. And, such features can be extracted by properly adding a domain loss to the loss function. Moreover, Figure-4.9 supports another point: the smaller domain loss means a closer distance between two domains. As we can tell from the figure, the ***Caltech-256 to Covid19-CT*** combination has the lowest domain loss, and it also has the best classification accuracy. Furthermore, the domain loss curve of Dslr data set increases during the training. It indicate that the quantity and the versatility of the source data set play an important role in this task. Finally, the author quantifies the

Table 4.10: DFF Performance

DFF	Accuracy	Precision	Recall	F1
Single Source	0.86	0.92	0.86	0.88
Multi-Source	0.88	0.92	0.93	0.92
Segmented Multi-Source	0.96	0.97	0.98	0.97

performance of DFF model with four evaluation metrics: accuracy, precision, recall, and F1 score.

4.2.6 Concluding Remark & Future Work

To draw a conclusion, in this study, the author introduces a novel DDTL framework (DFF) for medical imaging. Moreover, the author applies the proposed framework on COVID-19 diagnosis task to justify its proficiency. Moreover, the author conducts experiments with another 5 methods with different leaning manners: non-transfer, fine-tuning, DDTL (SLA). To distinguish our work from others, the proposed method can use seemingly unrelated data sets to develop an efficient classification model for COVID-19 diagnose. Unlike previous DDTL models, our method enables knowledge transfer from multiple distant source domains, and it can effectively enhance the performance on the COVID-19 diagnose. Moreover, the proposed method has great potential of expanding the usage of transfer learning on medical image processing by safely transferring the knowledge in distant source domains, which can be completely different from the target domain. Furthermore, this study is related to one of the most challenging problems in transfer learning, negative transfer. To the best of our knowledge, this is the first study that uses distant domain source data for COVID-19 diagnosis and outperforms promising test classification accuracy.

In addition, the framework is designed for general medical imaging tasks. COVID-19 diagnosis is just an example to justify the performance of the proposed work. However, the author also applies the framework to pneumonia diagnosis task. It also achieves decent performance (95.1 %) test classification accuracy. Intuitively, the reduced-UNet segmentation part is the key to improve the generalization ability of the framework. It is justified in [192] that the original UNet is effective for medical imaging tasks. Therefore, the framework can be extended to other medical imaging tasks by adjusting the size or the structure of the UNet based on the given data set. It proves that the proposed method has the ability of being adapted to other medical imaging methods. However, without the segmentation part, the proposed framework might also have the potential for regular image processing tasks. the author plans to conduct more research in the direction, but it is out of the scope of this study.

Four contributions of this study are made: 1) it successfully adopts DDTL methods to COVID-19 diagnosis, 2) the author introduces a novel feature-based DDTL classification algorithms, 3) the proposed methods achieve state-of-art results on COVID-19 diagnosis task, and 4) proposed methods can be easily expanded to other medical image processing problems.

However, there are several drawbacks of DDTL algorithms: 1) most algorithms tend to be case-specific, 2) source domain selection is too complicated in some cases, 3) distant feature extraction process is computationally expensive.

In the future, there are a number of research directions regarding COVID-19 diagnosis and DDTL problems. Firstly, the explainability of the feature-based DDTL algorithm is a challenging but essential topic. Visualizing the changes on features in deep layers through the training process can not only help us to better understand the domain adaptation in the feature level and decision making process of deep ANN models, but also discover the relationship between two distant domains. Moreover, how to improve the efficiency of feature extraction process is another key to improve the performance. Commonly, generative adversarial networks (GANs) is widely acknowledged as a better feature extraction method. However, how to avoid non-convergence in the training process of adversarial networks is very challenging, and gradient explode and disappear make the training process for adversarial networks extremely difficult. As an inspiration, designing new adversarial loss functions is a possible way of dealing with this problem. Moreover, there are many robust models pre-trained with large data sets, such as Resnet, Alexnet, and MDDA. Using pre-trained models as the feature extractors can significantly increase the distribution diversity of extracted features. However, it can lead to two major concerns: 1) re-training/fine-tuning such deep models is computationally expensive, and 2) increasing distribution diversity can cause over-fit. Therefore, dimensionality reduction and feature selection techniques can be the key to extend feature-based DDTL algorithms to large pre-trained models. Furthermore, cross-modality TL, such as from image to audio, can be another potential solution to DDTL problem since semantic information can also exist in different cross-modality domains. Solving this problem can expand the use of transfer learning to an even higher level. Furthermore, for multi-source DDTL algorithms, source domain selection is important to stabilize the performance. Recently, active learning methods attract more and more attention from researchers. Finally, using medical CT images from other diseases as the source domain might or might be able to produce better results because seemingly related domains can also have large discrepancies in the feature level. Moreover, image data sets are usually not easy to access, so it is not always feasible to develop a TL model by using medical image data from other diseases. Therefore, granting access to medical image data sets to the public and generating distribution shift embedded artificial data is a promising future research direction in the field of medical image processing.

Chapter 5

Cross-Modality Transfer Learning for Image-Text Information Management

In the past decades, information from all kinds of data has been on a rapid increase. With state-of-the-art performance, machine learning (ML) algorithms have been beneficial for information management. However, insufficient supervised training data is still an adversity in many real-world applications. Therefore, transfer learning (TF) was proposed to address this issue. This paper studies a not well-investigated but important TL problem termed Cross-Modality Transfer Learning (CMTL). This topic is closely related to distant domain transfer learning (DDTL) and negative transfer. In general, conventional TL disciplines assume that the source domain and the target domain are in the same modality. DDTL aims to make efficient transfers even when the domains or the tasks are entirely different. As an extension of DDTL, CMTL aims to make efficient transfers between two different data modalities, such as from image to text. As the main focus of this study, the author aims to improve the performance of image classification by transferring knowledge from text data. Previously, a few

CMTL algorithms were proposed to deal with image classification problems. However, most existing algorithms are very task-specific, and they are unstable on convergence. There are four main contributions in this study: 1) propose a novel heterogeneous CMTL algorithm, which requires only a tiny set of unlabeled target data and labeled source data with associate text tags, 2) introduce a latent semantic information extraction (LSIE) method to connect the information learned from the image data and the text data, 3) the proposed method can effectively handle the information transfer across different modalities (text-image), and 4) the author examined our algorithm on a public data set, Office-31. It has achieved up to 5% higher classification accuracy than ”non-transfer” algorithms and up to 9% higher than existing CMTL algorithms.

5.1 Introduction

In the past decades, the volume of information from all kinds of data modalities has increased rapidly. For example, with the modern internet system, a massive amount of image data can be accessed easily. However, a vast amount of redundant information can also be created, and it often gives us a hard time finding useful information. Therefore, it is essential to design more efficient and more effective information management methods that help us to extract useful information. In this paper, it focuses on improving the efficiency and the performance of image data management. Recently, machine learning has made breakthroughs in many different fields, including but not limited to image processing, speech recognition, and natural language processing (NLP). With state-of-art performances, machine learning models have been successfully applied to solve more and more real-world problems that traditional statistical learning methods cannot solve.

In general, traditional machine learning relies on a massive amount of training data. Moreover, it assumes one critical condition: the training data and the testing data are drawn from the same distribution. However, this assumption does not always hold in many real-world problems [197]. As such, most conventional machine learning algorithms usually suffer from three main difficulties: 1) insufficient data, 2) incompatible computation power, and 3) distribution mismatch. First of all, various solutions have been proposed to address the first two problems, such as data argumentation, data synthesis, distributed learning, and cloud computing. However, each of these proposed solutions is suffering from some adversities regarding high training cost, implementation efficiency, and the security. Recently, transfer learning (TL) has been brought to our attention to solve all three difficulties.

It has been proved that TL can handle all three problems in modern ML. Theoretically, transfer learning algorithms aim to develop robust target models by using only a small set of target training data and transferring knowledge learned from other domains and tasks. Recently, the modern TL has been extended to deep learning [58]. Moreover, the concept of adaptation layer with domain distance measurements was first proposed by [17]. It allows us to transfer knowledge between deep neural networks. In general, conventional transfer learning algorithms assume that the source domains and the targets share a certain amount of common information. However, this assumption does not always hold in many real-world applications, such as medical image processing [18, 19], rare species detection [20], and recommendation systems [21, 22]. In addition, transferring between two loosely related domains usually causes negative transfer [23–25], meaning that the knowledge transfer starts hurting the performance on the task in the target domain and produces worse performance than non-transfer models. For instance, building a dog classification model by directly transferring knowledge from a car clas- sification model will likely lead to negative transfer due to the weak connection between the two domains. Therefore, it is not always feasible to apply transfer learning to areas where we cannot easily obtain enough source domain data related to the target domain.

Previously, [77, 84] introduced a novel transfer learning discipline, Distant Domain Transfer Learning (DDTL). DDTL aims to address the issue of negative transfer caused by loose relations between the source domains and the target domains. In other words, it allows us to safely and effectively perform the knowledge transfer when the source domains and the target domains only share a very weak connection. The inspiration behind DDTL is that the ability of human beings to learn a new thing by using knowledge learned from several seemingly independent things. For example, a human who knows birds and airplanes can recognize a rocket even without seeing any rockets previously. Therefore, DDTL greatly extends the use of transfer learning to more areas, and applications there do not always have adequate related source data. Moreover, extracting domain-invariant features is challenging when the source domain and the target domain have a large domain discrepancy. Therefore, DDTL usually requires massive source data sets to extract a sufficient amount of meaningful and domain-invariant features. However, massive source data sets are not always accessible, and the computation cost is not always affordable.

Furthermore, DDTL can be further improved by embedding the information extracted from data sets in other modalities, such as image-text embedding. Image features cannot effectively represent semantic features in an image, and it is not easy to extract deep domain-invariant features with conventional TL. Therefore, the author proposes to improve the performance by using the semantic information provided by text tags as the side information. Moreover, it is easy to access images with some tags from websites like Wekipedia and flickr. Therefore, [81] first introduced a heterogeneous transfer learning framework for knowledge transfer between text and images. It observed that for a target-domain classification problem, some annotated images could be found on many social Web sites, which can serve as a bridge to transfer knowledge from the abundant text documents available over the Web. A critical issue for cross-modality information transfer is effectively converting the image information and the text information into the same format. It proposed to modify the representation of the target images with semantic concepts extracted from the auxiliary source data through a novel matrix factorization method by using the latent semantic features generated by the auxiliary data. However, it is not stable on convergence due to sparse matrix, and it relies on hand-designed image features.

In this paper, as inspired by the Neflix recommendation system [198], the author proposes a novel CMTL algorithm with a non-sparse semantic matrix to solve image classification tasks.

Moreover, the proposed algorithm makes two main improvements. Firstly, the proposed algorithm can deal with both labeled and unlabeled target domain data for domain adaptation problems. It can use a sufficient amount of labeled source domain data and some associate text tags to produce very promising classification accuracy on the target domain. Secondly, the proposed novel semantic information transformation method can avoid the sparse matrix. Moreover, the author applies a deep feature selection method, Distant Feature Fusion (DFF). It aims to discover general features across distant domains and tasks by using a convolutional autoencoder pair with a domain distance measurement. And then, the author introduces a novel latent semantic information extraction (LSIE) method. Furthermore, to justify the improvements, the author chooses a widely used public data set (Office-31) with manually added tags. With testing multiple benchmark models on the data set, the author shows that the proposed CMTL algorithm has achieved the highest accuracy on an image classification task. Compared with transfer learning methods, supervised learning methods, existing DDTL methods, and CMTL methods, our algorithm has achieved up to 9% higher classification accuracy.

The remainder of this paper is structured as follows: In Section 5.2, the author first reviews some related works. And then, Section 5.3 formulates the problem definition. After that, the details of the proposed algorithms are introduced in Section 5.4. Moreover, the author demonstrates experimental results and analysis in Section 5.5. Lastly, the author concludes the paper and discuss future directions in Section 5.6.

5.2 Related Work

Recently, insufficient training data and domain distribution mismatch have become the two most difficult ML challenges. As one of the solutions, TL has emerged more and more attention due to its training efficiency and domain shift robustness. In general, the conventional TL assumes that the source domain and the target domain are closely related [75]. However, this assumption does not hold in many real-world problems. A large domain discrepancy can cause negative transfer [23], which significantly limits the use and performance of TL. Recently, DDTL and Cross-Modality Transfer Learning (CMTL) have been proposed to address this issue. In this section, the author introduces some related works in three fields: conventional TL, DDTL, and CMTL.

First of all, TL aims to discover and transfer the domain-invariant and meaningful features in the source domain and the target domain. Originally, most TL algorithms focus on transferring knowledge with statistical and traditional models. More recently, [46] has expanded the use of TL from traditional ML models to deep neural networks. Typically, there are two types of TL algorithms: feature-based and instance-based. In common, both types aim to close the distribution distance between the source domain and the target domain. In instance-based algorithms, the goal is to discover source instances that are similar to target instances, so that the highly unrelated source samples would be eliminated. Instance-based methods require a massive amount of source data and computation power to select enough samples for the target task. Differently, feature-based algorithms aim to map source features and target features into a common feature space where the distribution mismatch is minimized. Feature-based methods usually require less source samples than instance-based methods. Importantly, both of them naturally assume that the source domain and the target domain are closely related. However, this assumption does not always hold since the distribution mismatch exists in many real-world problems. Furthermore, distant domains bring greater domain diversity which can lead to the issue of sparse domain-invariant features. Unlike conventional transfer learning, DDTL can transfer knowledge between different domains and tasks that are not closely related. Moreover, most DDTL algorithms are similar to multi-task learning [118], which also benefits from shared knowledge in multiple different but related domains. Generally, multi-task learning tends to improve the performance on all the tasks. Differently, DDTL only focuses on using the knowledge in other domains to improve the performance on the target task in the target domain.

Moreover, most previous studies of DDTL focus on instance-based methods and tend to take advantage of massive related source data. There were a few proposed instance-based DDTL algorithms [77, 79] previously. For example, the first study in this field was [77], transitive transfer learning (TTL). It transfers knowledge between text data in the source domain and the image data in the target domain using annotate image data as a bridge. However, this algorithm is highly case-dependent and unstable on performance. At a later time, [79] introduced another instance-based algorithm with a novel instance selection method, Selective Learning Algorithm (SLA). Moreover, it uses SLA to select helpful instances from a number of unrelated intermediate domains to expand the volume of the source domain. However, this algorithm was proposed to handle binary classification problems. Furthermore, [78] proposed another feature-based method to deal with scarce satellite image data. It predicts the poverty based on the daytime satellite image by transferring knowledge learned from object classification tasks with the help of some nighttime light intensity information as a bridge. However, this method has two major shortcomings. Firstly, it heavily relies on a massive amount of labeled intermediate training data, which can be too expensive to apply. Secondly, it cannot extract deep hidden features with the simple model architecture. Unlike existing DDTL algorithms, the proposed CMTL method can benefit from multiple source domains without labeled data, and those source domains can have significant discrepancies. Furthermore, our method can also handle multi-class classification and consistently produce promising results. Moreover, this study aims to use knowledge extracted from different data modalities to deal with sparse domain-invariant features.

Furthermore, CMTL is one of the most challenging topics in TL. It assumes that the source domain and the target domain share completely different spaces are, such as from text to image, from audio to text, and from image to audio. Moreover, the label spaces between the source and the target domain can also be different. Intuitively, CMTL is inspired by humans' ability to generalize knowledge from one subject to another by building a bridge with knowledge from other seemingly unrelated subjects. For example, a child who has read an article with descriptions of monkeys, and he has never seen any monkeys or images of monkeys. However, it is possible that the child can recognize a monkey based on the knowledge learned from the article. In this case, a child can transfer the knowledge from text data to image data using knowledge in other different domains. Theoretically, two seemingly unrelated domains can be connected by one or multiple bridge domains with overlapping semantic information. However, this type of learning behavior is counter-intuitive for machines to mimic due to the challenge in selecting appropriate intermediate domains as the bridge. Firstly, [80] researched heterogeneous transfer learning called Translated Learning via Risk Minimization (TLRisk). It proposed an asymmetric architecture to map the features in the source domain to the target domain. Moreover, it used a language model proposed by [93] and the nearest neighbor method to connect the text source data and the image target data. Moreover, to produce a smooth feature transition, it also developed a translator by applying the Markov chain. The source features and the target features were modeled by two different Markov chains bridged with intermediate data. In other words, the translation is done by learning a probabilistic model that uses cooccurrence data as a bridge between the source and target feature spaces. Finally, it proposed a variant of the risk minimization model to produce the final label prediction. This method conducted promising results that are better than the baseline model trained on only target data. However, the computational cost of TLRisk is very expensive due to the risk function estimation and dynamic programming. Differently, our CMTL algorithm uses the image data as the primary feature source and the text data as the secondary feature source. The text data aims to provide side semantic information to improve the image classification accuracy.

5.3 Problem Statement and Notation

In this section, the author introduces notations and give a clear problem statement.

5.3.1 Notation

As shown in Table 5.1, there are a number of frequently used notations throughout the chapter.

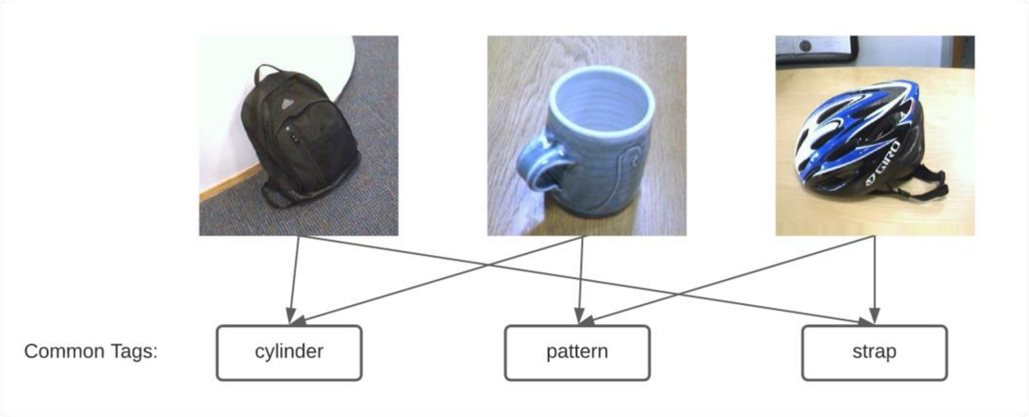


Figure 5.1: Three different objects share common semantic text information. Three seemingly unrelated images share common information in the text domain. For example, each image is associated with text tags. The backpack and the mug share the “cylinder” as the common information, the mug and the helmet share “strap”, and the backpack and the helmet share “pattern”.

Table 5.1: Notation

Term	Symbol
D_S	Source Domain
X_S	Source Domain Instance
Y_S	Source Domain Label Space
T_S	Source Domain Tag Space
D_T	Target Domain
X_T	Target Domain Instance

5.3.2 Problem Statement

In this CMTL problem, the author assumes that unlabeled target domain data $X_T = \{(x^1_T, x^2_T, ..., x^n_T)\}$ is not enough to train a robust model. However, there are a sufficient amount of labeled source domain data and a decent amount of text tags associated to the source domain (D_S) and the target domain (D_T). The source domain data is denoted as $X_S = \{(x^1_S, y^1_S), (x^2_S, y^2_S) ...., (x^n_S, y^n_S)\}$, and the associated tags are expressed as:

$$T_S = \{(x^1_S, t^1_S), (x^2_S, t^2_S)...., (x^n_S, t^n_S)\}, T_T = \{(x^1_T, t^1_T), (x^2_T, t^2_T)...., (x^n_T, t^n_T)\}. \tag{5.1}$$

Furthermore, the source domain and the target domain have a large distribution mismatch.

Let $P(x)$, $P(y|x)$ be the marginal and the conditional distributions of a data set. In this CMTL problem:

$$P_S(x) = P_T(x), \tag{5.2}$$

$$P_S(y|x) = P_T(y|x). \tag{5.3}$$

The proposed work aims to develop a model for the target domain with a minimal amount of unlabeled data by finding domain-invariant and meaningful features from distant unlabeled source domain data and combining latent semantic information extracted from text tags. The motivation behind this study is that data in distant domains is usually seemingly unrelated in instance-level but related on the feature-level. Moreover, as shown in Figure-5.1, different objects might share common latent semantic information in a different modality. For example, each image is associated with two text tags. The backpack and the mug share the "cylinder" as the common information, the mug and the helmet share "strap", and the backpack and the helmet share "pattern". In general, the connection on the feature level from one distant domain can be too weak to be used to train an accurate model, and the knowledge bridge between different modalities can be difficult to establish. As such, the two main challenges of this study are: 1) extracting distant features, and 2) bridging features extracted from different data modalities. In this study, the author proposes DFF and LSIE to solve the two challenges.

5.4 Methodology

In this section, the author introduces a novel heterogeneous CMTL algorithm, Distant Feature Fusion. As shown in Figure-5.2, there are three main components in our algorithm: distant feature fusion, latent semantic information fusion, and the target classification. The author gives details of each part in following sections.

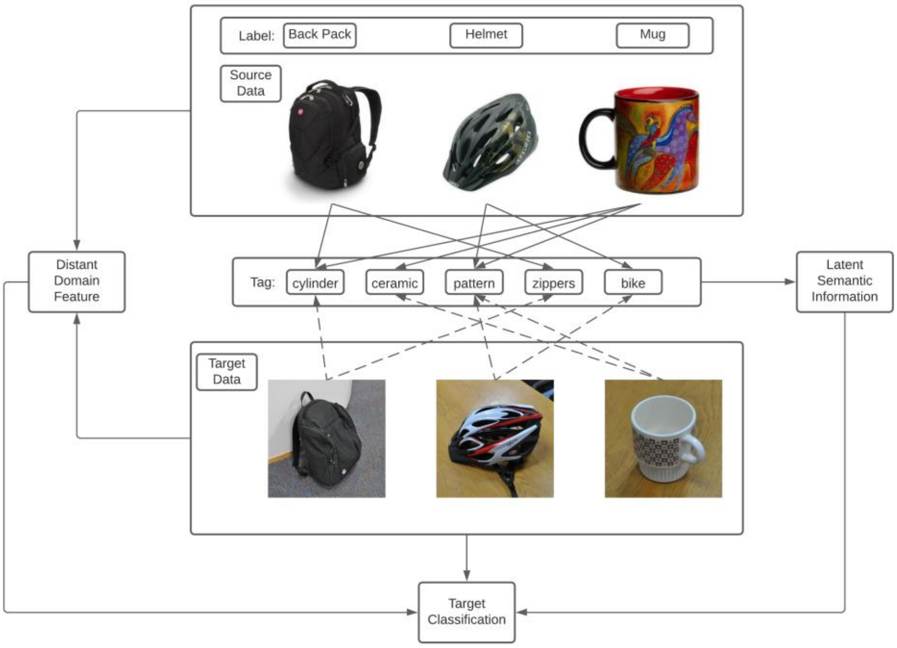


Figure 5.2: CMTL Architecture Overview: there are three main components in our algorithm: distant feature fusion, latent semantic information fusion, and the target classification.

5.4.1 Distant Feature Fusion

Primarily, there are no well-labeled target data or source data for the training process, so the feature extraction will follow the unsupervised manner. Therefore, a convolutional autoencoder pair is used as a feature extractor in DFF. As a variant of autoencoders, convolutional autoencoders [178] are usually beneficial to unsupervised image processing related problems. First of all, a convolutional autoencoder is a feed-forward neural network working in an unsupervised manner, which suits this problem perfectly since there is no labeled data in source domains. Generally, a convolutional autoencoder pair contains one input layer, one output layer, one up-sampling layer, and multiple convolutional layers. In general, there are two main components: encoder $E_{Conv}(\cdot)$ and decoder $D_{Conv}(\cdot)$. The standard process of convolutional autoencoder pairs can be demonstrated as:

$$Encoding : \boldsymbol{f} = E_{Conv}(x), Decoding : \hat{x} = D_{Conv}(\hat{f}), \tag{5.4}$$

where f is the extracted features of x, and $\hat{x}$ is the reconstruction of the original data sample. Furthermore, the way to tune the parameters of a convolutional autoencoder pair is to minimize the reconstruction error over all the training instances. Conceptually, the output of the encoder can be considered as high-level features of the unlabeled training data. Furthermore, these features are learned in an unsupervised manner, so they are robust when the reconstruction error is lower than a certain threshold. In other words, the encoder aims to discover a certain amount of representative features, and the decoder aims to ensure the extracted features are meaningful. Unlike supervised methods, this process does not require any labeled data.

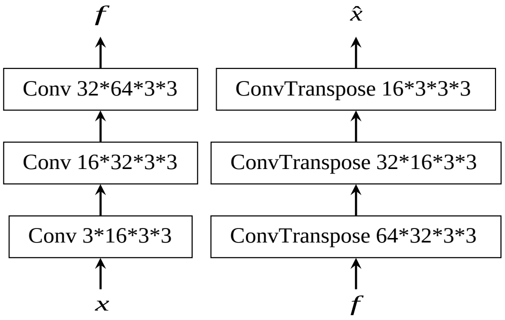


Figure 5.3: Encoder and Decoder: f is the extracted features of x, and $\hat{x}$ is the reconstruction of the original data sample.

In this problem, as shown in Figure 5.2, the author uses a pair of convolutional autoencoder to discover robust feature representation from unlabeled source domain data sets and the labeled target data sets simultaneously. Moreover, the structures of the encoder and the decoder can be found in Figure 5.3. There are two convolutional layers and two pooling layers in each of the encoder and the decoder. Moreover, up-sampling is applied to the encoder to ensure the quality of the reconstructed images. The process of feature extraction has three main steps: feature extraction, instance reconstruction, reconstruction measurement. First, both the source data and the target data are fed into the encoder to obtain high-level features f_S and f_T. And then, extracted features are sent into decoder to get reconstructions, $\hat{f}_S$ and $\hat{f}_T$. The equations of the first two steps are expressed as:

$$f_S = E_{Conv}(X_S), f_T = E_{Conv}(X_T); \tag{5.5}$$

$$\hat{X}_S = D_{Conv}(\boldsymbol{f}_S), \hat{X}_T = D_{Conv}(\boldsymbol{f}_T); \tag{5.6}$$

Finally, the reconstruction errors from both the source domains and the target domains are used to construct the loss function of the feature extractor, L_R is defined as:

$$L_R = \sum_{i=1}^{n} \frac{1}{n}(\hat{X}_{X_{S_i}} - X_{X_{S_i}})^2 + \sum_{i=1}^{m} \frac{1}{m}(\hat{X}_{X_{T_i}} - X_{X_{T_i}})^2. \tag{5.7}$$

Commonly, minimizing the reconstruction error L_R can discover a set of high-level features of the given input data. However, the distribution mismatch between the source and the target domains is significant, so minimizingL_R alone is not enough to extract robust and domain-invariant features. Therefore, extra side information can help us to close the domain distance, so the extracted features can be robust to both the source domains and the target domains. In this research, the author adds a distant feature adaptation layer to the convolutional autoencoder pair to measure the domain loss, L_D. The maximum mean discrepancy (MMD) [179], an important statistical domain distance estimator, is used as the domain distance measurement metric. The domain loss is expressed as:

$$L_D = MMD\left(\sum_{i=1}^{n} \boldsymbol{f}_{S_i}, \sum_{i=1}^{m} \boldsymbol{f}_{T_i}\right), \tag{5.8}$$

$$MMD(X, Y) = \left\| \frac{1}{n_1} \sum_{i=1}^{n_1} \phi(x_i) + \frac{1}{n_2} \sum_{f=1}^{n_2} \phi(y_j) \right\|, \tag{5.9}$$

where n_1 and n_2 are the numbers of instances of two different domains, and $\phi(\cdot)$ is the kernel that converts two sets of features to a common reproducing kernel Hilbert space (RKHS) where the distance of two domains is maximized. Furthermore, it allows us to extract a set of domain-invariant and meaningful features for the target classification. However, the extracted features might not be sufficient for developing a robust target classifier due to the larger discrepancy. Therefore, another set of additional information can ensure the performance of the target classifier.

```
Algorithm 4: Distant Feature Fusion Algorithm
Input: $S = X_S$, $T = X_T$ , $Y_T$ .
        Max Iteration: I, Batch Number: N.
for $i = 1, ...., I$ do
    for $j = 1, ...., N$ do
        Feature Extraction: $f_S = E_{Conv}(X_S)$, $f_T = E_{Conv}(X_T)$ Instance
         Reconstruction: $\hat{X}_S = D_{Conv}(X_S)$, $\hat{X}_T = D_{Conv}(X_S)$
         Label Prediction: $X^S_{Pred} = C_T(f_S)$
         Calculate $L_R$, $L_D$, $L_C$
         Update $\theta_E$, $\theta_D$, $\Theta_C$
    end
end
Output: $X^T_{Pred}$
```

5.4.2 Latent Semantic Information Extraction

To discover another set of additional information, the author wishes to take advantage of other data sets with different modalities. As mentioned earlier, seemingly unrelated images might share common information. In this study, there is an additional set of text tags associated to the source and the target images. Therefore, some additional information can be extracted from the text data to improve the performance of the target image classifier. Moreover, there are two major challenges of cross-modality information transfer: 1) cross-modality feature fusion and 2) high dimension vs. sparse matrix. In this section, the author introduces a method for latent semantic information extraction for image-text features. First of all, assume that there are totally h unique tags in the tag space T_S , and the instance-tag matrix $M_{IT} \in R^{n\times h}$, where n represents the total number of images. And then, the distant feature matrix can be found by feeding the image data into the DFF model. The distant feature matrix can be expressed as $M_{DFF} \in R^{n\times d}$, where d is the number of features of the last convolutional layer. After that, a tag-feature matrix is defined as: $M_{TF} = M^T_{DFF} M_{IT} \in R^{d\times h}$. In addition, this tag-feature matrix represents the correlation between image features and the text features. More importantly, M_{TF} is not sparse so that it can be effectively and safely decomposed at a later time. The intuitive reason behind it is that each element in the matrix is the cumulative value of a specific feature and the specific tag of all instances. Besides, the matrix can be visualized as:

$$M_{TF} = \begin{bmatrix} F_1T_1 & F_1T_2 & \dots\dots & F_1T_h \\ F_2T_1 & F_2T_2 & \dots. & F_2T_h \\ \dots & \dots & \dots. & \dots \\ \dots & \dots & \dots. & \dots \\ F_dT_1 & F_dT_2 & \dots\dots & F_dT_h \end{bmatrix}, \tag{5.10}$$

where each row represents the relation between a specific feature and all tags, and each column represents the relation between a specific tag and all features. Moreover, this matrix contains the information both from the image data and the text data. The next step is to discover a certain amount of latent semantic features from it.

Furthermore, motivated by Neflix Prize [198], the latent semantic information can be extracted by performing matrix decomposition:

$$M_{TF} = UV^T, \tag{5.11}$$

where $U \in R^{d\times l}$ and $V \in R^{h\times l}$, and l is the number of latent semantic features. In addition, l is a user-defined value which will be introduced with more details at a later time. Moreover, the author applies numerical optimization method for the matrix decomposition process and the loss is defined as:

$$L_{MD} = \| M_{TF} - \hat{U}\hat{V}^T \|^2 + \lambda R(\hat{U},\hat{V}), \tag{5.12}$$

where λ is the penalty coefficient, $R(\hat{U},\hat{V})$ is the penalty term to avoid over/under fitting, and $R(\hat{U},\hat{V}) = (\| \hat{U} \|^2 + \| \hat{V} \|^2)$.

Finally, with extracted latent semantic features, it can help us to reconstruct a new feature representation of the image data that embedded with both image features and text tag information. The new feature $F_{FT} = M_{DFF}U \in R^{n\times l}$. And then, the new set of features is used to train a new classifier constructed by two fully-connected layers.

5.4.3 Target Classifier

Furthermore, with new set of features, the author adds two fully-connected layers after the encoder to build a target classifier, C_T , for the target task in the target domain. As the motivation of this step, [10] proves that convolutional layers can discover features, and fully-connected layers can find the best feature combination for each class in the target task. In other words, fully-connected layers do not learn more new features but connect each class to a specific set of features with different weights. In this work, there is only one fully-connected layer followed by the output layer with cross-entropy loss, L_C:

$$L_C = -x[Class] + \mathop{}\limits_{i=1}^{n} exp(X_{S_i}). \tag{5.13}$$

Finally, by embedding all three losses from 5.7, 5.8, and 5.13, the overall objective function of DFF is formulated as:

$$\underset{\theta_E,\theta_D,\Theta_C}{\text{Minimize}} \quad L = L_R + L_D + L_C, \tag{5.14}$$

where θ_E, θ_D, Θ_C are the parameters of the encoder, decode, and the classifier, respectively. Moreover, L is the final loss constructed by the reconstruction error, domain loss, and classification loss. Finally, all the parameters are optimized by minimizing the objective function in Equation 5.14. However, the classification loss is designed for the final target classification, and it is optional in the distant feature extraction process. With or without it would not significantly vary the results. More details will be discussed this at a later time. Moreover, the overview of DFF is summarized in Algorithm 4.

5.5 Experiment and Analysis

In this section, the author first introduces the data set and experimental setups. And then, the author compares our algorithm with a number of benchmark models, such as supervised learning models, conventional transfer learning models, DDTL models and CMTL models. After that, the author presents results from the proposed CMTL model and the comparisons with benchmark models. Finally, the author demonstrates details of the proposed algorithm and result analysis.

5.5.1 Data Set

In this study, the author chooses a widely used public data set, Office-31 [183], which has three collections of total 4110 instances from three different data sources: “amazon”, “webcam”, and “dslr”. Moreover, the author randomly selects 10 classes to manually add $1-5$ text tags to each sample. Moreover, as shown in Figure-5.1, text tags describe the appearance, the shapes, or the functions of each object. Moreover, the author performs three knowledge transfers in each experiment, namely “*amazon*”*to*“*webcam*”, “*webcam*”*to*“*amazon*”, “*webcam*”*to*“*dslr*”, “*webcam*”*to*“*dslr*”, “*amazon*”*to*“*dslr*”, and “*dslr*”*to*“*amazon*”.

5.5.2 Bench Mark Model

Firstly, the author selects one non-transfer supervised baseline models: pre-trained ResNet50 [76]. And then, the author picks two conventional transfer learning models: Manifold Dynamic Distribution Adaptation - ReNet (MDDA) [199] and Multi-Adversarial Domain Adaptation - AlexNet (MADA) [200]. Moreover, the author chooses one instance-based DDTL algorithm: selective learning algorithm (SLA) [79]. Lastly, the author selects another CMTL algorithm: Heterogeneous Transfer Learning for Image Classification (HTLIC) [81].

5.5.3 Performance and Analysis

First of all, the author runs each experiment five times to obtain each method’s performance variation range. As shown in Table 5.2, with insufficient labeled training data, non-transfer methods still carry out fairly decent testing classification accuracy (78.3%). Conventional

Table 5.2: Accuracy (%) of Experiments on Office-31

	ResNet-50	MDDA	MADA	SLA	HTLIC	Ours (no tags)	Ours (with tags)
amazon-webcam	82 ± 0.2	84 ± 0.1	75 ± 0.4	88 ± 2	60 ± 0.2	85 ± 0.2	**89 ± 0.1**
webcam-amazon	64 ± 0.1	**71 ± 0.3**	47 ± 0.2	68 ± 3	40 ± 0.1	61 ± 0.3	67 ± 0.1
webcam-dslr	92 ± 0.1	**98 ± 0.1**	90 ± 0.4	62 ± 3	65 ± 0.2	88 ± 0.1	97 ± 0.1
dslr-webcam	91 ± 0.3	94 ± 0.2	91 ± 0.1	88 ± 2	63 ± 0.2	86 ± 0.2	**96 ± 0.2**
amazon-dslr	77 ± 0.2	85 ± 0.1	76 ± 0.5	68 ± 3	54 ± 0.3	74 ± 0.4	**89 ± 0.1**
dslr-amazon	64 ± 0.1	71 ± 0.1	57 ± 0.2	62 ± 3	45 ± 0.2	64 ± 0.2	**77 ± 0.1**
Average	78.3	83.8	72.6	62	54.5	76.3	**85.3**

transfer learning algorithms are able to bypass the accuracy achieved by the ResNet50 model. However, MDDA (83.8%) is the only conventional method that is better than the non-transfer model in this study. The MADA model is only at 72.6%, which is not promising. Moreover, the DTTL algorithm (SLA) outputs the second-worst performance (62%), which is not much better than a pure guess. After that, the previous CMTL method (HTLIC) has the worst accuracy (54.5%), which is a case of negative transfer. Finally, the proposed algorithm carries out decent performance (76.3%) without using tag information. With tag information, the performance is dramatically improved to 85.3%, which the highest in all tested methods. In addition, our algorithm achieves the best performance in four settings: "*amazon*"*to*"*webcam*", "*webcam*"*to*"*dslr*", "*amazon*"*to*"*dslr*", and "*dslr*"*to*"*amazon*".

Moreover, Figure-5.4 illustrates the domain distance changing through the training and demonstrates that the final domain distance is closely related to the latent semantic information extracted from text tags. The domain distance is a lot smaller when the semantic features are added to the training process. Furthermore, it also approves that the distant feature adaption layer can close the distribution mismatch even when domains are very distant. What is more, as illustrated in Figure-5.5, the number of latent semantic features can greatly affect the performance and convergence time. As we can tell, the performance first goes up as the number of semantic features increases, then it hits the peak at 50 semantic features and starts decreasing after. In general, more information should help the model to learn more useful knowledge. However, way too many features can also involve noise which can hurt the performance. In this study, 50 latent semantic features yield the best accuracy. In addition, the convergence time keeps increasing when the author adds more latent semantic features, and the model does not converge with more than 100 latent semantic features.

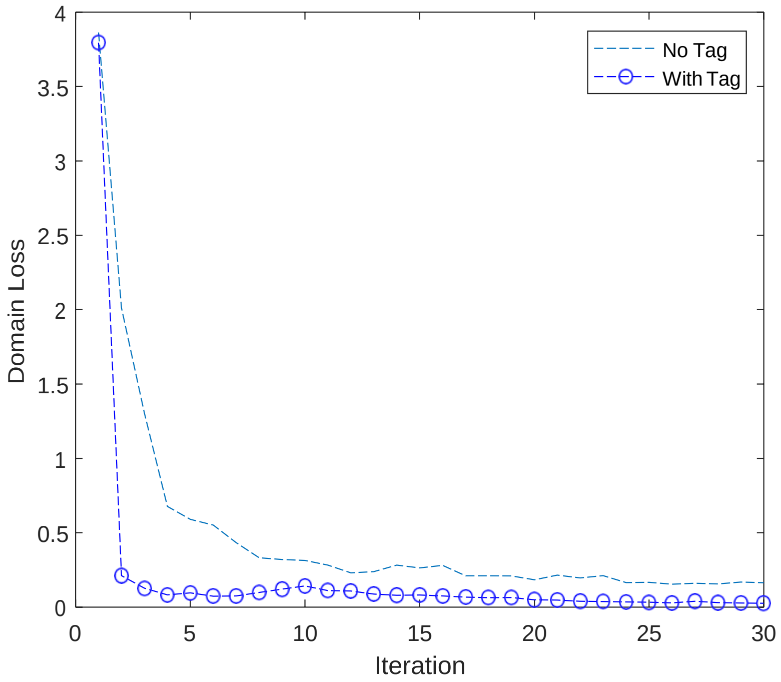


Figure 5.4: The domain loss: the final domain distance is closely related to the latent semantic information extracted from text tags. The domain distance is a lot smaller when the semantic features are added to the training process.

5.6 Concluding Remarks

In this chapter, the author studies a CMTL problem in image-text information management, where only exists a decent amount of labeled source data with text tags and a small set of unlabeled target domain data collected from very distant domains and tasks. Under this setting, conventional transfer learning algorithms usually suffer from negative transfer. The author introduces a novel heterogeneous CMTL algorithm to address this problem, which can effectively extract and fuse the distant features learned from distant domains and latent semantic features from different data modalities. Unlike other ML algorithms, CMTL can handle multiple source and target domains, and it does not rely on any labeled data from the target domain. Moreover, DFF can achieve effective distant feature extraction, and LSIE can discover semantic information across modalities. Furthermore, the author also conducts a series of experiments on Office-31 and present an analysis of the proposed algorithm.

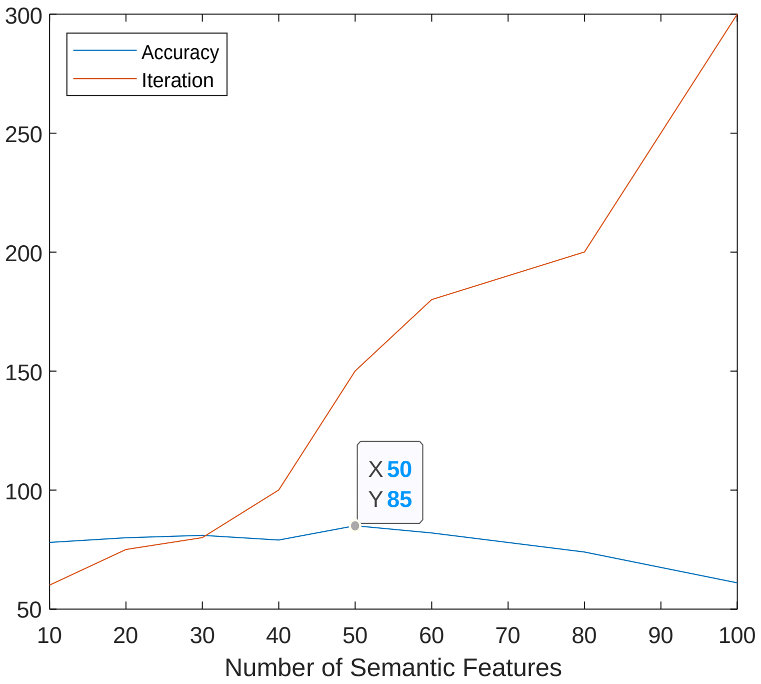


Figure 5.5: The impact of the number of semantic features: the performance first goes up as the number of semantic features increases, then it hits the peak at 50 semantic features and starts decreasing after.

In the future, there are two directions regarding this CMTL problem. Firstly, the explainability of the CMTL algorithm is a challenging but essential problem. Visualizing the changes in high-level features through the training process can help us understand the domain adaptation on the feature level. Secondly, how to effectively set the number of latent semantic features is another challenge. The iterative method used in this study is fairly computationally expensive and inefficient. Solving this problem can expand the use of CMTL algorithms to an even further level.

Chapter 6

Concluding Remark and Future Work

6.1 Concluding Remark

In section-3, a well-defined DEML framework and a product-oriented evaluation system. The DEML framework covers the most commonly used methods to ease the performance degradation caused by insufficient training data and incompatible computation power. In addition, these methods are organized into two main categories: 1) data science and 2) learning algorithms. In data science, there are two highlighted methods, data augmentation, and data re-sampling. In data augmentation, the author introduces a series of methods that can effectively increase training data volume. These methods can be implemented in several areas, such as image processing, audio analysis, and signal processing. Besides generating artificial data, the author also presents methods that can maximize the use of existing data sets. There are three widely used techniques in data re-sampling: 1) validation, 2) cross-validation, and 3) bootstrap, which can efficiently re-use small data sets. However, data science methods cannot provide more distribution diversity to original data sets. Therefore, improvements in learning algorithms are needed when the data volume gets to a certain level.

There are three learning disciplines in learning algorithms: FSL, ensemble learning, and TL, respectively. Additionally, TL is the main focus of this dissertation at the algorithm-level:

- Few-shot learning benefits from meta-learning, and it can carry out decent results by using only a few or zero data samples. Under this setting, it might not always learn directly from the target samples. Instead, it might learn some features from other samples that are related to the target.
- Ensemble learning performs well with a small data set by combing multiple weak learners. It assumes that weaker learners do not require a massive data set for training. The more common ensemble learning method is random forests.
- As the main focus of this dissertation, TL aims to solve the target task by transferring knowledge learned from other domains, so it does not need to learn from scratch with a massive amount of data.

More importantly, TL has been successfully adapted to deep learning. However, conventional TL assumes that the source domain and target domains are closely related. Negative transfer occurs when there is a large discrepancy between two domains. This dissertation proposes two novel algorithms to avoid negative transfer.

Recently, TL has been successfully applied to many real-world problems that traditional machine learning (ML) cannot handle, such as image processing, speech recognition, and natural language processing (NLP). Commonly, TL tends to address three main problems of traditional machine learning: (1) insufficient labeled data, (2) incompatible computation power, and (3) distribution mismatch. In general, TL can be organized into four categories: transductive learning, inductive learning, unsupervised learning, and negative learning. Each category can be organized into four learning types: learning on instances, learning on features, learning on parameters, and learning on relations. This article presents a comprehensive survey on TL. Besides, this chapter presents the state of the art, current trends, applications, and open challenges. In section-2, it presents an updated survey by demonstrating the state-of-the-art, current trends and open challenges in the field. While most recent surveys equally cover mainstream TL topics, our survey extends that by identifying and discussing the most challenging TL problems, such as distant domain and cross-modality TL. The survey promotes the positive applications of transfer learning to foster a broader community in the field.

Moreover, in section-3, TL techniques are applied to a real-world application, solid waste sorting. A novel loss function, the Dual Dynamic Domain Loss function (4D), is introduced to provide more accurate domain distance measurements. And then, as mentioned earlier, how to address the negative transfer issue when transferring knowledge among distant domains is a key to expand the use of TL. Therefore, in section-4, the author proposes a novel feature-based DDTL algorithm to negate the negative transfer between distant domains. This topic is closely related to negative transfer. Unlike conventional transfer learning problems, DDTL aims to make efficient transfers when the domains or the tasks are completely different. Most existing algorithms are very task-specific, and they are instance-based. This study proposed a feature-based algorithm that requires only a tiny set of labeled target data and unlabeled source data from completely different domains. Instead of selecting intermediate instances, the author develops Distant Feature Fusion (DFF), a novel feature selection method, to discover general features cross distant domains and tasks. As the novelty of this study, it can effectively handle both distant domain multi-class image classification and binary image classification problems. Furthermore, this DDTL algorithm is applied to medical imaging in section-4.

As an extension of DDTL, CMTL is another very important but not well-studied TL problem. DDTL aims to make efficient transfers even when the domains or the tasks are entirely different. As an extension of DDTL, CMTL aims to make efficient transfers between two different modalities, such as from image to text. As the main focus of this study, the author aims to improve the performance of image information classification by transferring knowledge between text data and image data. Previously, a few CMTL algorithms were proposed to deal with image classification problems. However, most existing algorithms are very task-specific, and they are unstable on convergence. There are four main contributions in this study: 1) propose a novel heterogeneous CMTL algorithm, which requires only a tiny set of unlabeled target data and labeled source data with associate text tags, 2) introduce a latent semantic information extraction (LSIE) method to connect the information learned from the image data and the text data, 3) the proposed method can effectively handle the information transfer across different modalities (text-image), and 4) the author examines the proposed algorithm on a public data set, Office-31. It has achieved promising performance.

6.2 Future Works

As the future plan, there are two promising directions: 1) TL algorithms 2) applied-TL. In this section, the author will discuss the details in each direction.

6.2.1 TL Algorithms

Many studies of TL have carried out promising performances in several fields. However, there are still some open challenges that are waiting to be addressed. Moreover, there are four research directions in algorithms:

- Human-Guided TL: enable the model to master a task from scratch without any human experiences and instructions.
- Negative Transfer: it occurs when the distribution mismatch between two domains is large. It is always the case in real-world problems. For example, a promising direction is discover methods that can efficiently transfer knowledge from pre-trained deep models, such as ResNet and MDDA.
- Adversarial TL: adversarial models are generally more powerful but difficult to train. It is not well-investigated in transfer learning.
- Transfer Learning with Graph Neural Networks: graph neutral networks is a newly proposed concept. It has the potential in transfer learning.

For human-guided TL, the key is how to provide human pre-experience to the learning models correctly. With such pre-experience, the training process of TL models can be even more efficient. Moreover, to negate negative, there are three potential solutions, such as DDTL, CMTL, and meta-TL. In addition, DDTL and meta-TL are very similar concepts, and they can be combined by using active learning methods. For CMTL, as extensions of the proposed CMTL algorithm, future works can focus on transferring with other data modalities, such as video and audio. Besides, adversarial-based TL is more powerful, but it is more difficult to train due to non-convergence. As future plans, designing new training protocols that can perform stabilize the training. Furthermore, in real-world situations, we are dealing with spatial-temporal data sets that are not euclidean-based. Therefore, TL with traditional learning methods is not suitable for such tasks. As such, developing graph neural networks-based TL models is also very important.

6.2.2 TL Applications

Moreover, applying algorithms to practical problems can help us build a bridge between theories and reality. There are several areas that can greatly benefit from TL techniques:

- Computer Vision: medical image processing (classification and segmentation), diagnosis assistant (image-based detection and image enhancement).
- Natural Language Processing: text semantic analysis, workflow analysis
- Smart and Connected Community: smart home, activity recognition, indoor location, transportation with emergency response.
- Others: recommendation system, personalized treatment planing, human-machine interface.

Bibliography

[1] Rui Xia, Xuelei Hu, Jianfeng Lu, Jian Yang, and Chengqing Zong. Instance selection and instance weighting for cross-domain sentiment classification via pu learning. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[2] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*, 2014.

[3] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 1180–1189. JMLR.org, 2015. URL http://dl.acm.org/citation.cfm?id=3045118.3045244.

[4] Yang Wu, Wei Li, Michihiko Minoh, and Masayuki Mukunoki. Can feature-based inductive transfer learning help person re-identification? In *2013 IEEE international conference on image processing*, pages 2812–2816. IEEE, 2013.

[5] Zhongtang Zhao, Yiqiang Chen, Junfa Liu, Zhiqi Shen, and Mingjie Liu. Cross-people mobile-phone based activity recognition. In *Twenty-second international joint conference on artificial intelligence*, 2011.

[6] Lilyana Mihalkova, Tuyen Huynh, and Raymond J Mooney. Mapping and revising markov logic networks for transfer learning. In *Aaai*, volume 7, pages 608–614, 2007.

[7] Lilyana Mihalkova and Raymond J Mooney. Transfer learning by mapping with minimal target data. In *Proceedings of the AAAI-08 workshop on transfer learning for complex tasks*, 2008.

[8] Jesse Davis and Pedro Domingos. Deep transfer via second-order markov logic. In *Proceedings of the 26th annual international conference on machine learning*, pages 217–224. ACM, 2009.

[9] Guo-Jun Qi, Charu Aggarwal, and Thomas Huang. Towards semantic knowledge propagation from text corpus to web images. In *Proceedings of the 20th international conference on World wide web*, pages 297–306, 2011.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[11] Mingsheng Long, Yue Cao, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Transferable representation learning with deep adaptation networks. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[12] S. Niu, Y. Liu, J. Wang, and H. Song. A decade survey of transfer learning (2010-2020). *IEEE Transactions on Artificial Intelligence*, pages 1–1, 2021. doi: 10.1109/TAI.2021.3054609.

[13] Neil C Thompson, Kristjan Greenewald, Keeheon Lee, and Gabriel F Manso. The computational limits of deep learning. *arXiv preprint arXiv:2007.05558*, 2020.

[14] Shyam Patidar, Dheeraj Rane, and Pritesh Jain. A survey paper on cloud computing. In *2012 Second International Conference on Advanced Computing & Communication Technologies*, pages 394–398. IEEE, 2012.

[15] Weisong Shi, Jie Cao, Quan Zhang, Youhuizi Li, and Lanyu Xu. Edge computing: Vision and challenges. *IEEE internet of things journal*, 3(5):637–646, 2016.

[16] Yuan Yu Michael Isard Dennis Fetterly, Mihai Budiu, Úlfar Erlingsson, and Pradeep Kumar Gunda Jon Currey. Dryadlinq: A system for general-purpose distributed data-parallel computing using a high-level language. *Proc. LSDS-IR*, 8, 2009.

[17] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

[18] Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016.

[19] Veronika Cheplygina, Marleen de Bruijne, and Josien PW Pluim. Not-so-supervised: a survey of semi-supervised, multi-instance, and transfer learning in medical image analysis. *Medical image analysis*, 54:280–296, 2019.

[20] Jaclyn N Taroni, Peter C Grayson, Qiwen Hu, Sean Eddy, Matthias Kretzler, Peter A Merkel, and Casey S Greene. Multiplier: a transfer learning framework for transcriptomics reveals systemic features of rare disease. *Cell systems*, 8(5):380–394, 2019.

[21] Weike Pan, Evan Wei Xiang, Nathan Nan Liu, and Qiang Yang. Transfer learning in collaborative filtering for sparsity reduction. In *Twenty-fourth AAAI conference on artificial intelligence*, 2010.

[22] Weike Pan, Nathan N Liu, Evan W Xiang, and Qiang Yang. Transfer learning to predict missing ratings via heterogeneous user feedbacks. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[23] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[24] Bishwaranjan Bhattacharjee, John R Kender, Matthew Hill, Parijat Dube, Siyu Huo, Michael R Glass, Brian Belgodere, Sharath Pankanti, Noel Codella, and Patrick Watson. P2l: Predicting transfer learning for images and semantic relations. In *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, pages 760–761, 2020.

[25] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[26] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010.

[27] Fabio Maria Cariucci, Lorenzo Porzi, Barbara Caputo, Elisa Ricci, and Samuel Rota Bulò. Autodial: Automatic domain alignment layers. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5077–5085. IEEE, 2017.

[28] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

[29] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *arXiv preprint arXiv:1911.02685*, 2019.

[30] Oscar Day and Taghi M Khoshgoftaar. A survey on heterogeneous transfer learning. *Journal of Big Data*, 4(1):29, 2017.

[31] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer, 2018.

[32] John E Ball, Derek T Anderson, and Chee Seng Chan. Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11(4):042609, 2017.

[33] Ankit Narendrakumar Soni. Application and analysis of transfer learning-survey. *International Journal of Scientific Research and Engineering Development*, 1(2):272–278, 2018.

[34] Ruijun Liu, Yuqian Shi, Changjiang Ji, and Ming Jia. A survey of sentiment analysis based on transfer learning. *IEEE Access*, 7:85401–85412, 2019.

[35] Weike Pan. A survey of transfer learning for collaborative recommendation with auxiliary data. *Neurocomputing*, 177:447–453, 2016.

[36] Abu Sufian, Anirudha Ghosh, Ali Safaa Sadiq, and Florentin Smarandache. A survey on deep transfer learning to edge computing for mitigating the covid-19 pandemic. *Journal of Systems Architecture*, 108:101830, 2020.

[37] Jie Lu, Vahid Behbood, Peng Hao, Hua Zuo, Shan Xue, and Guangquan Zhang. Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems*, 80: 14–23, 2015.

[38] Qingyao Wu, Hanrui Wu, Xiaoming Zhou, Mingkui Tan, Yonghui Xu, Yuguang Yan, and Tianyong Hao. Online transfer learning with multiple homogeneous or heterogeneous sources. *IEEE Transactions on Knowledge and Data Engineering*, 29(7):1494–1507, 2017.

[39] Muhammad Ghifary, W Bastiaan Kleijn, and Mengjie Zhang. Domain adaptive neural networks for object recognition. In *Pacific Rim international conference on artificial intelligence*, pages 898–904. Springer, 2014.

[40] Rui Xia, Chengqing Zong, Xuelei Hu, and Erik Cambria. Feature ensemble plus sample selection: domain adaptation for sentiment classification. *IEEE Intelligent Systems*, 28 (3):10–18, 2013.

[41] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[42] Artem Rozantsev, Mathieu Salzmann, and Pascal Fua. Beyond sharing weights for deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 41 (4):801–814, 2019.

[43] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2272–2281, 2017.

[44] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

[45] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.

[46] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2208–2217. JMLR. org, 2017.

[47] Yanghao Li, Naiyan Wang, Jianping Shi, Xiaodi Hou, and Jiaying Liu. Adaptive batch normalization for practical domain adaptation. *Pattern Recognition*, 80:109–117, 2018.

[48] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

[49] Feng Xu, Jianfei Yu, and Rui Xia. Instance-based domain adaptation via multiclustering logistic approximation. *IEEE Intelligent Systems*, 33(1):78–88, 2018.

[50] Rui Xia, Jianfei Yu, Feng Xu, and Shumei Wang. Instance-based domain adaptation in nlp via in-target-domain logistic approximation. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[51] Yi Yao and Gianfranco Doretto. Boosting for transfer learning with multiple sources. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1855–1862. IEEE, 2010.

[52] Hal Daumé III. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*, 2009.

[53] Durjoy Sen Maitra, Ujjwal Bhattacharya, and Swapan K Parui. Cnn based common approach to handwritten character recognition of multiple scripts. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1021–1025. IEEE, 2015.

[54] Min Fang, Yong Guo, Xiaosong Zhang, and Xiao Li. Multi-source transfer learning based on label shared subspace. *Pattern Recognition Letters*, 51:101–106, 2015.

[55] Muhammad Jamal Afridi, Arun Ross, and Erik M Shapiro. On automated source selection for transfer learning in convolutional neural networks. *Pattern recognition*, 73:65–75, 2018.

[56] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[57] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[58] Shuteng Niu, Jian Wang, Yongxin Liu, and Houbing Song. Transfer learning based data-efficient machine learning enabled classification. In *2020 IEEE Intl Conf on Dependable,*

Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), pages 620–626. IEEE, 2020.

[59] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 193–200, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: 10.1145/1273496.1273521. URL http://doi.acm.org/10.1145/1273496.1273521.

[60] Yoav Freund, Robert Schapire, and Naoki Abe. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

[61] G. Matasci, D. Tuia, and M. Kanevski. Svm-based boosting of active learning strategies for efficient domain adaptation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(5):1335–1343, Oct 2012. ISSN 1939-1404. doi: 10.1109/JSTARS.2012.2202881.

[62] Z. Li, B. Liu, and Y. Xiao. Cluster and dynamic-tradaboost-based transfer learning for text classification. In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 2291–2295, July 2017. doi: 10.1109/FSKD.2017.8393128.

[63] Z. Yuan, D. Bao, Z. Chen, and M. Liu. Integrated transfer learning algorithm using multi-source tradaboost for unbalanced samples classification. In *2017 International Conference on Computing Intelligence and Information System (CIIS)*, pages 188–195, April 2017. doi: 10.1109/CIIS.2017.37.

[64] Samir Al-Stouhi and Chandan K. Reddy. Adaptive boosting for transfer learning using dynamic updates. In *Proceedings of the 2011 European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part I*, ECML PKDD'11, pages 60–75, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-23779-9. URL http://dl.acm.org/citation.cfm?id=2034063.2034080.

[65] David Pardoe and Peter Stone. Boosting for regression transfer. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pages 863–870, USA, 2010. Omnipress. ISBN 978-1-60558-907-7. URL http://dl.acm.org/citation.cfm?id=3104322.3104432.

[66] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and S Yu Philip. Learning multiple tasks with multilinear relationship networks. In *Advances in neural information processing systems*, pages 1594–1603, 2017.

[67] Wan-Yu Deng, Qing-Hua Zheng, and Zhong-Min Wang. Cross-person activity recognition using reduced kernel extreme learning machine. *Neural Networks*, 53:1–7, 2014.

[68] Huan Li, Yuan Shi, Yang Liu, Alexander G Hauptmann, and Zhang Xiong. Cross-domain video concept detection: A joint discriminative and generative active learning approach. *Expert Systems with Applications*, 39(15):12220–12228, 2012.

[69] Fabian Nater, Tatiana Tommasi, Helmut Grabner, Luc Van Gool, and Barbara Caputo. Transferring activities: Updating human behavior analysis. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1737–1744. IEEE, 2011.

[70] Zheng Wang, Yangqiu Song, and Changshui Zhang. Transferred dimensionality reduction. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 550–565. Springer, 2008.

[71] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Self-taught clustering. In *Proceedings of the 25th international conference on Machine learning*, pages 200–207. ACM, 2008.

[72] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: Transfer learning from unlabeled data. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 759–766, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: 10.1145/1273496.1273592. URL http://doi.acm.org/10.1145/1273496.1273592.

[73] Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, and Vincent Vanhoucke. Youtube-boundingboxes: A large high-precision human-annotated data set for object detection in video. pages 7464–7473, 07 2017. doi: 10.1109/CVPR.2017.789.

[74] L. Duan, D. Xu, and S. Chang. Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1338–1345, June 2012. doi: 10.1109/ CVPR.2012.6247819.

[75] Liang Ge, Jing Gao, Hung Ngo, Kang Li, and Aidong Zhang. On handling negative transfer and imbalanced distributions in multiple source transfer learning. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 7(4):254–271, 2014.

[76] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[77] Ben Tan, Yangqiu Song, Erheng Zhong, and Qiang Yang. Transitive transfer learning. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1155–1164, 2015.

[78] Michael Xie, Neal Jean, Marshall Burke, David Lobell, and Stefano Ermon. Transfer learning from deep features for remote sensing and poverty mapping. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[79] Ben Tan, Yu Zhang, Sinno Jialin Pan, and Qiang Yang. Distant domain transfer learning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[80] Wenyuan Dai, Yuqiang Chen, Gui-Rong Xue, Qiang Yang, and Yong Yu. Translated learning: Transfer learning across different feature spaces. In *Advances in neural information processing systems*, pages 353–360, 2009.

[81] Yin Zhu, Yuqiang Chen, Zhongqi Lu, Sinno Jialin Pan, Gui-Rong Xue, Yong Yu, and Qiang Yang. Heterogeneous transfer learning for image classification. In *AAAI*, volume 11, pages 1304–1309, 2011.

[82] Qingyao Wu, Michael K Ng, and Yunming Ye. Cotransfer learning using coupled markov chains with restart. *IEEE intelligent Systems*, 29(4):26–33, 2013.

[83] Michael K Ng, Qingyao Wu, and Yunming Ye. Co-transfer learning via joint transition probability graph based method. In *Proceedings of the 1st international workshop on cross domain knowledge discovery in web and social network mining*, pages 1–9, 2012.

[84] Shuteng Niu, Hu Yihao, Jian Wang, Yongxin Liu, and Houbing Song. Feature-based distant domain transfer learning. In *2020 IEEE International Conference on Big Data*. IEEE, 2020.

[85] Joshua Lee, Prasanna Sattigeri, and Gregory W. Wornell. Learning new tricks from old dogs: Multi-source transfer learning from pre-trained networks. In *NeurIPS*, 2019.

[86] CB Bell. Mutual information and maximal correlation as measures of dependence. *The Annals of Mathematical Statistics*, pages 587–595, 1962.

[87] Boyu Wang, Jorge Mendez, Mingbo Cai, and Eric Eaton. Transfer learning via minimizing the performance gap between domains. In *Advances in Neural Information Processing Systems*, pages 10644–10654, 2019.

[88] John Blitzer, Ryan McDonald, and Fernando Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128. Association for Computational Linguistics, 2006.

[89] Y. Yao and G. Doretto. Boosting for transfer learning with multiple sources. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1855–1862, June 2010. doi: 10.1109/CVPR.2010.5539857.

[90] Eric Eaton et al. Selective transfer between learning tasks using task-based boosting. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[91] Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 264–271, 2007.

[92] Xuejun Liao, Ya Xue, and Lawrence Carin. Logistic regression with an auxiliary data source. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, pages 505–512, New York, NY, USA, 2005. ACM. ISBN 1-59593-180-5. doi: 10.1145/1102351.1102415. URL http://doi.acm.org/10.1145/1102351.1102415.

[93] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*, 2018.

[94] Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3): 273–297, 1995.

[96] Raymond T Ng and Jiawei Han. Efficient and effective clustering methods for spatial data mining. In *Proceedings of VLDB*, pages 144–155, 1994.

[97] Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer, 2018.

[98] Yuan Yuan, Xiangtao Zheng, and Xiaoqiang Lu. Hyperspectral image superresolution by transfer learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(5):1963–1974, 2017.

[99] Kasthurirangan Gopalakrishnan, Siddhartha K Khaitan, Alok Choudhary, and Ankit Agrawal. Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and Building Materials*, 157:322–330, 2017.

[100] Mostafa Mehdipour Ghazi, Berrin Yanikoglu, and Erchan Aptoula. Plant identification using deep neural networks via optimization of transfer learning parameters. *Neurocomputing*, 235:228–235, 2017.

[101] Keunwoo Choi, György Fazekas, Mark Sandler, and Kyunghyun Cho. Transfer learning for music classification and regression tasks. *arXiv preprint arXiv:1703.09179*, 2017.

[102] Ye Jia, Yu Zhang, Ron Weiss, Quan Wang, Jonathan Shen, Fei Ren, Patrick Nguyen, Ruoming Pang, Ignacio Lopez Moreno, Yonghui Wu, et al. Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In *Advances in neuralinformation processing systems*, pages 4480–4490, 2018.

[103] Anurag Kumar, Maksim Khadkevich, and Christian Fügen. Knowledge transfer from weakly labeled audio using convolutional neural network for sound events and scenes. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 326–330. IEEE, 2018.

[104] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[105] Xin Luna Dong and Gerard De Melo. A helping hand: Transfer learning for deep sentiment analysis. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2524–2534, 2018.

[106] Jyoti Islam and Yanqing Zhang. Visual sentiment analysis for social images using transfer learning approach. In *2016 IEEE International Conferences on Big Data and*

Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom), pages 124–130. IEEE, 2016.

[107] Siddique Latif, Rajib Rana, Shahzad Younis, Junaid Qadir, and Julien Epps. Transfer learning for improving speech emotion classification accuracy. *arXiv preprint arXiv:1801.06353*, 2018.

[108] Joey Tianyi Zhou, Sinno Jialin Pan, Ivor W Tsang, and Shen-Shyang Ho. Transfer learning for cross-language text categorization through active correspondences construction. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[109] Giang Pham, Danaee Donovan, Quynh Dam, and Amy Contant. Learning words and definitions in two languages: What promotes cross-language transfer? *Language learning*, 68(1):206–233, 2018.

[110] Luis HS Vogado, Rodrigo MS Veras, Flavio HD Araujo, Romuere RV Silva, and Kelson RT Aires. Leukemia diagnosis in blood slides using transfer learning in cnns and svm for classification. *Engineering Applications of Artificial Intelligence*, 72:415–422, 2018.

[111] Rich Colbaugh, Kristin Glass, and Gil Gallegos. Ensemble transfer learning for alzheimer's disease diagnosis. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3102–3105. IEEE, 2017.

[112] Benjamin Q Huynh, Hui Li, and Maryellen L Giger. Digital mammographic tumor classification using transfer learning from deep convolutional neural networks. *Journal of Medical Imaging*, 3(3):034501, 2016.

[113] Nicholas Siekirk, Qin Lai, and Bradley Kendall. Effects of limb-specific fatigue on motor learning during an upper extremity proprioceptive task. *International Journal of Motor Control and Learning*, 1(1):76–81, 2018.

[114] Jay Lee, Moslem Azamfar, Jaskaran Singh, and Shahin Siahpour. Integration of digital twin and deep learning in cyber-physical systems: towards smart manufacturing. *IET Collaborative Intelligent Manufacturing*, 2(1):34–36, 2020.

[115] Tingting Hou, Gang Feng, Shuang Qin, and Wei Jiang. Proactive content caching by exploiting transfer learning for mobile edge computing. *International Journal of Communication Systems*, 31(11):e3706, 2018.

[116] Susanne Loidl. Towards pervasive learning: Welearn. mobile. a cps package viewer for handhelds. *Journal of Network and ComputerApplications*, 29(4):277–293, 2006.

[117] Han Zou, Yuxun Zhou, Hao Jiang, Baoqi Huang, Lihua Xie, and Costas Spanos. Adaptive localization in dynamic indoor environments by transfer kernel learning. In *2017 IEEE wireless communications and networking conference (WCNC)*, pages 1–6. IEEE, 2017.

[118] Wenlu Zhang, Rongjian Li, Tao Zeng, Qian Sun, Sudhir Kumar, Jieping Ye, and Shuiwang Ji. Deep model based transfer and multi-task learning for biological image analysis. *IEEE transactions on Big Data*, 2016.

[119] Yong Xu, Jun Du, Li-Rong Dai, and Chin-Hui Lee. Cross-language transfer learning for deep neural network based speech enhancement. In *The 9th International Symposium on Chinese Spoken Language Processing*, pages 336–340. IEEE, 2014.

[120] Yadunath Pathak, Prashant Kumar Shukla, Akhilesh Tiwari, Shalini Stalin, Saurabh Singh, and Piyush Kumar Shukla. Deep transfer learning based classification model for covid-19 disease. *IRBM*, 2020.

[121] Ioannis D Apostolopoulos and Tzani A Mpesiana. Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks. *Physical and Engineering Sciences in Medicine*, page 1, 2020.

[122] Mohamed Loey, Florentin Smarandache, and Nour Eldeen M Khalifa. Within the lack of chest covid-19 x-ray dataset: A novel detection model based on gan and deep transfer learning. *Symmetry*, 12(4):651, 2020.

[123] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018. URL http://science.sciencemag.org/content/362/6419/1140/tab-pdf.

[124] Wei Ying, Yu Zhang, Junzhou Huang, and Qiang Yang. Transfer learning via learning to transfer. In *International Conference on Machine Learning*, pages 5085–5094, 2018.

[125] Ping Li, Jin Li, Zhengan Huang, Tong Li, Chong-Zhi Gao, Siu-Ming Yiu, and Kai Chen. Multi-key privacy-preserving deep learning in cloud computing. *Future Generation Computer Systems*, 74:76–85, 2017.

[126] Dashan Gao, Yang Liu, Anbu Huang, Ce Ju, Han Yu, and Qiang Yang. Privacy-preserving heterogeneous federated transfer learning. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 2552–2559. IEEE, 2019.

[127] Qinbin Li, Zeyi Wen, and Bingsheng He. Federated learning systems: Vision, hype and reality for data privacy and protection. *arXiv preprint arXiv:1907.09693*, 2019.

[128] Shiwei Zhang, Xiuzhen Zhang, Jeffrey Chan, and Paolo Rosso. Irony detection via sentiment-based transfer learning. *Information Processing & Management*, 56(5):1633–1644, 2019.

[129] Jiang-Jing Lv, Xiao-Hu Shao, Jia-Shui Huang, Xiang-Dong Zhou, and Xi Zhou. Data augmentation for face recognition. *Neurocomputing*, 230:184–196, 2017.

[130] Z. Zhong, L. Zheng, Z. Zheng, S. Li, and Y. Yang. Camstyle: A novel data augmentation method for person re-identification. *IEEE Transactions on Image Processing*, 28(3): 1176–1190, March 2019. ISSN 1057-7149. doi: 10.1109/TIP.2018.2874313.

[131] Xiang Wang, Kai Wang, and Shiguo Lian. A survey on face data augmentation, 2019.

[132] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):60, 2019.

[133] X. Cui, V. Goel, and B. Kingsbury. Data augmentation for deep neural network acoustic modeling. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5582–5586, May 2014. doi: 10.1109/ICASSP.2014.6854671.

[134] Justin Salamon and Juan Pablo Bello. Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Processing Letters*, 24 (3):279–283, 2017.

[135] U. Aftab and G. F. Siddiqui. Big data augmentation with data warehouse: A survey. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2785–2794, Dec 2018. doi: 10.1109/BigData.2018.8622206.

[136] David A Van Dyk and Xiao-Li Meng. The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10(1):1–50, 2001.

[137] Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel. Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems*, pages 396–404, 1990.

[138] Francisco J Moreno-Barea, Fiammetta Strazzera, José M Jerez, Daniel Urda, and Leonardo Franco. Forward noise adjustment scheme for data augmentation. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 728–734. IEEE, 2018.

[139] Guoliang Kang, Xuanyi Dong, Liang Zheng, and Yi Yang. Patchshuffle regularization. *ArXiv*, abs/1707.07103, 2017.

[140] Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. Between-class learning for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5486–5494, 2018.

[141] Yuji Tokozume, Yoshitaka Ushiku, and Tatsuya Harada. Learning from between-class examples for deep sound recognition. *arXiv preprint arXiv:1711.10282*, 2017.

[142] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017.

[143] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur. Audio augmentation for speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[144] Raul Fernandez, Andrew Rosenberg, Alexander Sorin, Bhuvana Ramabhadran, and Ron Hoory. Voice-transformation-based data augmentation for prosodic classification. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5530–5534. IEEE, 2017.

[145] Xiaodong Cui, Vaibhava Goel, and Brian Kingsbury. Data augmentation for deep neural network acoustic modeling. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 23(9): 1469–1477, September 2015. ISSN 2329-9290. doi: 10.1109/TASLP.2015.2438544. URL http://dx.doi.org/10.1109/TASLP.2015.2438544.

[146] Anton Ragni, Katherine Mary Knill, Shakti P Rath, and Mark John Gales. Data augmentation for low resource languages. 2014.

[147] Teng Zhang, Kailai Zhang, and Ji Wu. Data independent sequence augmentation method for acoustic scene classification. In *Interspeech*, pages 3289–3293, 2018.

[148] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[149] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/90e1357833654983612fb05e3ec9148c-Paper.pdf.

[150] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/cb8da6767461f2812ae4290eac7cbc42-Paper.pdf.

[151] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[152] Houbing Song, Ravi Srinivasan, Tamim Sookoor, and Sabina Jeschke. *Smart Cities: Foundations, Principles and Applications*. Wiley, Hoboken, NJ, 2017. ISBN 978-1-119-22639-0.

[153] G Thippa Reddy, M Praveen Kumar Reddy, Kuruva Lakshmanna, Dharmendra Singh Rajput, Rajesh Kaluri, and Gautam Srivastava. Hybrid genetic algorithm and a fuzzy logic classifier for heart disease diagnosis. *Evolutionary Intelligence*, 13(2):185–196, 2020.

[154] Houbing Song, Danda Rawat, Sabina Jeschke, and Christian Brecher. *Cyber-Physical Systems: Foundations, Principles and Applications*. Academic Press, Boston, MA, 2016. ISBN 978-0-12-803801-7.

[155] Guido Dartmann, Houbing Song, and Anke Schmeink. *Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things*. Elsevier, 2019. ISBN 9780128166376.

[156] G Thippa Reddy, M Praveen Kumar Reddy, Kuruva Lakshmanna, Rajesh Kaluri, Dharmendra Singh Rajput, Gautam Srivastava, and Thar Baker. Analysis of dimensionality reduction techniques on big data. *IEEE Access*, 8:54776–54788, 2020.

[157] Y. Sun, H. Song, A. J. Jara, and R. Bie. Internet of things and big data analytics for smart and connected communities. *IEEEAccess*, 4:766–773, 2016. doi: 10.1109/ACCESS.2016.2529723.

[158] Z. Lv, H. Song, P. Basanta-Val, A. Steed, and M. Jo. Next-generation big data analytics: State of the art, challenges, and future research topics. *IEEE Transactions on Industrial Informatics*, 13(4):1891–1899, 2017. doi: 10.1109/TII.2017.2650204.

[159] Adel Ghorani-Azam, Bamdad Riahi-Zanjani, and Mahdi Balali-Mood. Effects of air pollution on human health and practical measures for prevention in Iran. *Journal of research in medical sciences: the official journal of Isfahan University of Medical Sciences*, 21, 2016.

[160] Mindy Yang and Gary Thung. Classification of trash for recyclability status. *CS229 Project Report*, 2016, 2016.

[161] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

[162] X. Yue, Y. Liu, J. Wang, H. Song, and H. Cao. Software defined radio and wireless acoustic networking for amateur drone surveillance. *IEEE Communications Magazine*, 56(4):90–97, April 2018. doi: 10.1109/MCOM.2018.1700423.

[163] Rahmi Arda Aral, Seref Recep Keskin, Mahmut Kaya, and Murat Haciomeroglu. Classification of trashnet dataset based on deep learning models. *2018 IEEE International Conference on Big Data (Big Data)*, pages 2058–2062, 2018.

[164] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861, 2017. URL http://arxiv.org/abs/1704.04861.

[165] Sebastian Ruder. An overview of gradient descent optimization algorithms. *CoRR*, abs/1609.04747, 2016. URL http://arxiv.org/abs/1609.04747.

[166] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014. URL http://arxiv.org/abs/1409.4842.

[167] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *J. Mach. Learn. Res.*, 13(1):723–773, March 2012. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=2503308.2188410.

[168] Kaggle Team. Classify waste category from images, 2018. URL https://www.kaggle.com/c/waste-classification/overview.

[169] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[170] Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems*, pages 1205–1213, 2012.

[171] Y. Liu, X. Weng, J. Wan, X. Yue, H. Song, and A. V. Vasilakos. Exploring data validity in transportation systems for smart cities. *IEEE Communications Magazine*, 55(5):26–33, 2017. doi: 10.1109/MCOM.2017.1600240.

[172] J. Yang, C. Wang, B. Jiang, H. Song, and Q. Meng. Visual perception enabled industry intelligence: State of the art, challenges and prospects. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2020. doi: 10.1109/TII.2020.2998818.

[173] Y. Liu, J. Wang, J. Li, H. Song, T. Yang, S. Niu, and Z. Ming. Zero-bias deep learning for accurate identification of internet of things (iot) devices. *IEEE Internet of Things Journal*, pages 1–1, 2020. doi: 10.1109/JIOT.2020.3018677.

[174] Yongxin Liu, Jian Wang, Shuteng Niu, and Houbing Song. Deep learning enabled reliable identity verification and spoofing detection. In Dongxiao Yu, Falko Dressler, and Jiguo Yu, editors, *Wireless Algorithms, Systems, and Applications - 15th International Conference, WASA 2020, Qingdao, China, September 13-15, 2020, Proceedings, Part I*, volume 12384 of *Lecture Notes in Computer Science*, pages 333–345. Springer, 2020. doi: 10.1007/978-3-030-59016-1\ 28. URL https://doi.org/10.1007/978-3-030-59016-1_28.

[175] S. Niu, J. Wang, Y. Liu, and H. Song. Transfer learning based data-efficient machine learning enabled classification of trashnet. In *6th IEEE International Conference on Cloud and Big Data Computing (IEEE CBDCom 2020)*, 2020.

[176] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014.

[177] Weifeng Ge and Yizhou Yu. Borrowing treasures from the wealthy: Deep transfer learning through selective joint fine-tuning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1086–1095, 2017.

[178] Volodymyr Turchenko, Eric Chalmers, and Artur Luczak. A deep convolutional autoencoder with pooling-unpooling layers in caffe. *arXiv preprint arXiv:1701.04949*, 2017.

[179] Karsten M Borgwardt, Arthur Gretton, Malte J Rasch, Hans-Peter Kriegel, Bernhard Schölkopf, and Alex J Smola. Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57, 2006.

[180] Lipo Wang. *Support vector machines: theory and applications*, volume 177. Springer Science & Business Media, 2005.

[181] Yves Grandvalet and Stéphane Canu. Adaptive scaling for feature selection in svms. In *Advances in neural information processing systems*, pages 569–576, 2003.

[182] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. *Non*, 2007.

[183] Yunhan Zhao, Haider Ali, and René Vidal. Stretching domain adaptation: How far is too far? *ArXiv*, abs/1712.02286, 2017.

[184] Sabina Jeschke, Christian Brecher, Houbing Song, and Danda Rawat. *Industrial Internet of Things*. Springer, Cham, Switzerland, 2017. ISBN 978-3-319-42558-0.

[185] Y. Zhang, L. Sun, H. Song, and X. Cao. Ubiquitous wsn for healthcare: Recent advances and future prospects. *IEEE Internet of Things Journal*, 1(4):311–318, 2014. doi: 10.1109/JIOT.2014.2329462.

[186] S. Niu, Y. Hu, J. Wang, Y. Liu, and H. Song. Feature-based distant domain transfer learning. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 1–8, 2020.

[187] Shuteng Niu, Jian Wang, Yongxin Liu, and Houbing Song. Transfer learning based Data-Efficient machine learning enabled classification. In *The 6th International Conference on Cloud and Big Data Computing (2020) (CBDCom 2020)*, August 2020.

[188] Y Liu, J Wang, S Niu, and H Song. (2020) deep learning enabled reliable identity verification and spoofing detection. 2020.

[189] Xuehai He, Xingyi Yang, Shanghang Zhang, Jinyu Zhao, Yichen Zhang, Eric Xing, and Pengtao Xie. Sample-efficient deep learning for covid-19 diagnosis based on ct scans. *medrxiv*, 2020.

[190] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[191] Mucahid Barstugan, Umut Ozkaya, and Saban Ozturk. Coronavirus (covid-19) classification using ct images by machine learning methods. *arXiv preprint arXiv:2003.09424*, 2020.

[192] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[193] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[194] Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5):1122–1131, 2018.

[195] Samuel G Armato III, Geoffrey McLennan, Luc Bidaut, Michael F McNitt-Gray, Charles R Meyer, Anthony P Reeves, Binsheng Zhao, Denise R Aberle, Claudia I Henschke, Eric A Hoffman, et al. The lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans. *Medical physics*, 38(2):915–931, 2011.

[196] Jinyu Zhao, Yichen Zhang, Xuehai He, and Pengtao Xie. Covid-ct-dataset: a ct scan dataset about covid-19. *arXiv preprint arXiv:2003.13865*,2020.

[197] Youcef Djenouri, Jerry Chun-Wei Lin, Kjetil Nørvåg, Heri Ramampiaro, and Philip S. Yu. Exploring decomposition for solving pattern mining problems. *ACM Trans. Manage. Inf. Syst.*, 12(2), February 2021. ISSN 2158-656X. doi: 10.1145/3439771. URL https://doi.org/10.1145/3439771.

[198] James Bennett, Stan Lanning, et al. The netflix prize. In *Proceedings of KDD cup and workshop*, volume 2007, page 35. New York, 2007.

[199] Jindong Wang, Yiqiang Chen, Wenjie Feng, Han Yu, Meiyu Huang, and Qiang Yang. Transfer learning with dynamic distribution adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(1):1–25, 2020.

[200] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. *arXiv preprint arXiv:1809.02176*, 2018.