EXPANSE: A Deep Continual / Progressive Learning System for Deep Transfer Learning

by

Mohammadreza Iman

(Under the Direction of Hamid R. Arabnia)

Abstract

The artificial intelligence (AI) community has one final goal, Artificial General Intelligence (AGI). A human-like intelligence with precision and processing speed of computers. Deep Learning (DL) has been the answer to many machine learning problems during the past two decades. However, it comes with two major constraints: dependency on extensive labeled data and training costs. Transfer learning in deep learning, known as Deep Transfer Learning (DTL), attempts to reduce such dependency and costs by reusing an obtained knowledge from a source data/task in training on a target data/task. Like any new advancement, DTL methods have their own limitations. The current DTL techniques suffer from either catastrophic forgetting dilemma (losing the previously obtained knowledge) or overly biased pre-trained models (harder to adapt to target data) in finetuning pre-trained models or freezing a part of the pre-trained model, respectively. Progressive learning, a sub-category of DTL, reduces the effect of the overly biased model in the case of freezing earlier layers by adding a new layer to the end of a frozen pre-trained model. Even though it has been successful in many cases, it cannot yet handle distant source and target data.

We propose a new continual/progressive learning approach for deep transfer learning to tackle these limitations. We expand the pre-trained model by expanding pre-trained layers (adding new nodes to each layer) in the model instead of only adding new layers. Hence the method is named EXPANSE. Our experimental results confirm that we can tackle distant source and target data using this technique. At the same time, the final model is still valid on the source data, achieving a promising deep continual learning approach. Moreover, we offer a new way of training deep learning models inspired by the human education system. We termed this two-step training: learning basics first, then adding complexities and uncertainties. The evaluation implies that the two-step training extracts more meaningful features and a finer basin on the error surface since it can achieve better accuracy in comparison to regular training. EXPANSE (model expansion and two-step training) is a systematic continual learning approach applicable to different problems and DL models.

INDEX WORDS: Artificial General Intelligence, Deep Transfer Learning, Deep Learning, Continual Learning, Progressive Learning, Transfer Learning

EXPANSE: A Deep Continual / Progressive Learning System for Deep Transfer Learning

by

Mohammadreza Iman

B.Sc., University of Shiraz, Iran, 2013

M.Sc., Federal University of Rio de Janeiro, Brazil, 2015

A Dissertation Submitted to the Graduate Faculty of the

University of Georgia in Partial Fulfillment of the Requirements for the Degree

Doctor of Philosophy

Athens, Georgia

2022

EXPANSE: A Deep Continual / Progressive Learning System for Deep Transfer Learning

by

Mohammadreza Iman

Major Professor: Hamid R. Arabnia

Committee: John A. Miller
Khaled Rasheed
Robert M. Branch

Electronic Version Approved:

Ron Walcott
Vice Provost for Graduate Education and
Dean of the Graduate School
The University of Georgia
August 2022

Dedication

With sincere gratitude and warm regard, I dedicate this work to my dear wife, family, friends, teachers, and advisors. I have drawn inspiration from each of them in a unique way that has helped me achieve my goals.

Acknowledgments

It is my honor to be an Iranian descendant of Cyrus the Great. In accordance with Zoroastrianism (the ancient Persian religion), Cyrus the Great adhered to the rules of "Good Thoughts, Good Words, and Good Deeds." He declared 2,500 years ago that "he would not reign over the people if they did not want it." It was his promise not to force anyone to change their religion and faith, as well as his guarantee of freedom for all. Accordingly, first of all, I thank God for giving me the wisdom, strength, support, and knowledge to explore, grow, and learn.

In spite of the fact that only my name appears on the cover of this dissertation, there are many people who have contributed to my PhD journey. This dissertation would not have been possible without many people who made it possible and helped me to complete my PhD program. I am grateful to all who have made this journey possible; because of this, my PhD degree experience has been one that I will cherish forever.

I have been blessed by having the most experienced, knowledgeable, and proficient advisor and committee members in my PhD program at UGA: Professor Hamid Reza Arabnia (Computer science Professor Emeritus), Professor John Miller (Computer science Professor and Graduate Coordinator), Professor Khaled Rasheed (Computer Science Professor and Director of the Institute for Artificial Intelligence), and Professor Robert Branch (Professor of Learning, Design, and

Technology and the Associate Head of the Department of Career and Information Studies). I treasure having access to such competent professors and receiving their directions and advice.

My deepest gratitude is to my advisor, Professor Arabnia. I have been amazingly fortunate to have an advisor who gave me the freedom to explore on my own and, at the same time, the guidance to recover when my steps faltered. His patience, advice, and support helped me overcome many baffling situations in my academic and personal life during the past five years. I hope that one day I will become as professional and wise as him.

Professor Miller always encouraged me in my research with his neat ideas and directions. His energetic and enthusiastic approach to teaching and in the meetings always motivates you to pursue new ideas and learn the topics more profoundly. One of my best memories of classes at UGA is his Data Science course, in which we decided to take the challenge of simulating a deep learning model via Excel to understand every step of calculations in training the model.

I started my PhD program with Professor Rasheed's machine learning class in my first semester at UGA. I learned a vast majority of my knowledge in the machine learning area from his class and his directions later in research that I have done under his supervision. Without the knowledge gained in Professor Rasheed's class, I would not have been able to conduct this research.

My decision to do the minor in Education was confirmed when I met Professor Branch in his instructional design course. His systematic instructional design methodology taught me not only how to design courses but also how to do research, prepare presentations, write reports, and even deal with personal life challenges. I have been honored by his acceptance to be my PhD minor committee. Since then, his kindness and endless support in every step of my PhD program were the most heartwarming and motivational experiences for me.

I would like to express my appreciation to Professor Thiab Taha. It would not have been possible for any of these to take place without his vision and partnership with the computer science department (now the school of computing).

I also want to express my gratitude to all my professors, teachers, and UGA's staff in the computer science department, education department, and graduate school. Last but not least, I am thankful for my great friends, classmates, and colleagues who supported me on this journey. The list of the people who helped me last five years is endless, as well as my appreciation towards them.

Above all, I want to thank my dear wife, family, and friends for their continued support during my doctorate program.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

More than seven decades ago, Alan Turing started Artificial Intelligence (AI) by asking the big question: "Can machines think?" [1]. The journey toward Artificial General Intelligence (AGI) started then and still is the main goal of the AI community [2]. Early AI models were based on considering all the possible actions and reactions and programming such details, e.g., checkers-playing (1951). Such an approach is only practical for already solved problems. However, most of the problems in the real world are a combination of uncertainties. Therefore, later the era of Machine Learning (ML) started to deal with uncertainty and address unsolvable problems. [2]

Until two decades ago, most ML algorithms could not deal with non-linear data, which is the most common type of data in daily problems. However, the rise of Deep Learning (DL) in the early 2000s opened the door to dealing with uncertainties and non-linear data, even outperformed humans in some tasks [2]. Another significant achievement in the growth path of AIs

was the bloom of Convolutional Neural Networks (CNN) around a decade ago, which drastically improved the DL models' accuracy on image-related data and tasks [3-2].

Even though Deep Learning (DL) brought a high accuracy and deals with many complex data and tasks, it comes with its own constraints. (i) **DL is highly dependent on abundant training data** (most cases labeled data), which is often impossible to harvest or is very costly. (ii) **The training process is very costly** in the matter of time and processing power.

Since the development of traditional machine learning algorithms in the 1980s, Transfer Learning has been a thrilling field of research for scientists. Transfer Learning (TL) is about using obtained knowledge from a source domain/data to facilitate learning on a target domain/data, also known as Domain Adaptation [3]. Traditional ML models are less dependent on extensive training data; the training process is more straightforward than Deep Learning (DL) models. Therefore, transfer learning in deep learning, known as Deep Transfer Learning (DTL) [4], is much more in demand and has been used vastly in the last decade to deal with DL's two aforementioned significant constraints

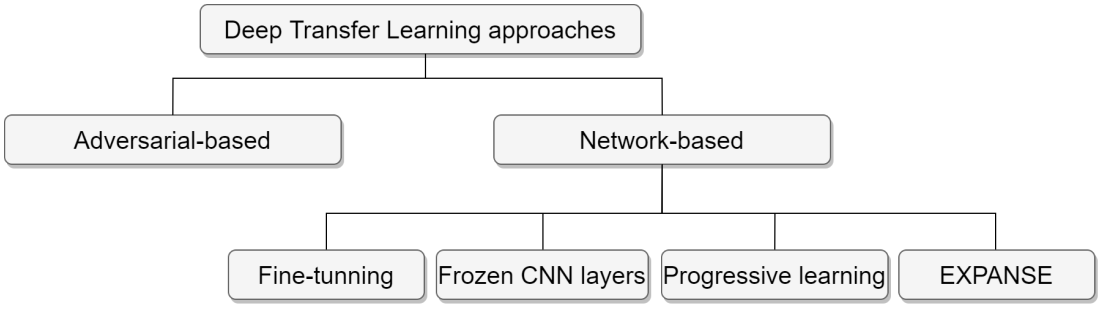

Figure 1.1: Deep transfer learning common approaches.

Deep Transfer Learning (DTL) methods are mostly Network/Model-based: (i) finetuning a pre-trained model; (ii) freezing a part of a pre-trained model and finetuning the rest; and (iii) progressive learning. Another approach in DTL is Adversarial-based, which is not as common as

model-based approaches [4]. Our proposed methodology, EXPANSE, is also a network/model-based approach. Figure 1.1 shows the hierarchy of common approaches in DTLs and the EXPANSE.

Depending on the approach from finetuning to freezing layers, DTL suffers from **catastrophic forgetting** or an **overly biased model**. The catastrophic forgetting dilemma happens while a pre-trained model trains on target data; in most cases, the previously obtained knowledge is partially or entirely wiped out [5], [6] . The other end of this spectrum is when several layers of a pre-trained model are frozen to train on target data. In this case, the pre-trained model is overly biased on source data and will not adapt to target data. A consequence of these two constraints is that we can not reach continual learning through current DTLs. In continual learning, a model should learn new skills while still handling the old skill(s).

Progressive learning / Progressive Neural Network (PNN) [7], [8] was introduced by the Google Deep Mind project in 2016, which is the closest attempt to mimic the human ability of continual learning (building a new skill on top of a previously learned skill). PNN has been applied to various problems, such as natural language processing and image-related tasks, successfully since 2016 [9]–[12]. In PNN, they freeze the pre-trained model on source data and train on the target data; they add a new layer(s) to the end of the model (near the output). Still, the method is overly biased on source data. It cannot deal with distant source and target data since the earlier layers are frozen (no learning capacity for additional detailed features). However, it can deal with task transfer better than the other approaches because of new lateral layers [8].

The final goal of Artificial General Intelligence (AGI) and the AI community is to imitate human intelligence and the learning process. Human intelligence and wisdom are the results of lifelong learning and training. We build knowledge on top of previously learned knowledge in

continual/progressive processes. E.g., we learn about simple shapes and objects (rectangles, circles, squares, etc.). As we grow using that fundamental knowledge, we learn more complex shapes and objects (cube, octagonal, airplane, car, etc.) and how to distinguish them in a different context [13].

We start to learn fundamentals in our childhood through exemplary (perfected) and simplified examples. We started learning shapes and letters by practicing basic examples in pre-school and kindergarten. We add to our knowledge and skills during K-12 school, building on what we previously learned. As our knowledge grows, we learn about more complex and ambiguous problems in a continuous learning process. A vital point of this learning process is that we start with the exemplary (perfect) and straightforward fundamental knowledge to reach wisdom after long-life training [13]. Inspired by this learning process, we introduce two-step training of deep learning models in EXPANSE.

Training processes in machine learning and deep learning are usually based on the training dataset's mixed distribution (mostly normal distribution). Deep learning models are initialized by random weights and start extracting features from details to abstract (first layers towards output layer) [14] with no prior knowledge. An extreme analogy of this regular training process would be asking a child to detect cancer tissues after showing him/her a thousand images of cancer tissues.

We have almost no idea what features the model is extracting in this training format and if those are valid for similar yet slightly different datasets. In other words, the extracted features could be strongly related to the specific training samples and not based on any fundamental knowledge. For instance, detecting cancer tissues could have been done by the shape of the area and not based on discoloration or texture of tissue. This is why a trained model for detecting animals can be

deceived by changing the context/background in the image while a human can detect it in different contexts.

EXPANSE introduces a new system of continual learning for deep learning inspired by the human education system and the current progressive learning technique. The first aspect of this proposed system is to change the process of training deep learning models based on the human learning process, **two-step training**. The second is to **expand the network while expanding the knowledge in the vertical dimension by adding new nodes on pre-trained layers**. We also consider that we may need to expand the network horizontally for some transfers, similar to how progressive learning adds new layers to the model. In deep learning, earlier layers extract detailed features, and the lateral layers towards the output extract more abstract knowledge using the extracted detailed features [14]. In EXPANSE, we offer to expand layers by adding new nodes to them to increase the model learning capacity. We show that in this way, not only can the final model deal with the target data; it is still valid on source data, which opens a reliable path to reach **continual learning**.

This dissertation is written based on our published papers of [2], [4], [13], and [15].

1.2 Outline

The rest of the dissertation is organized as follows.

Chapter 2: This chapter covers the background and basics of Artificial Intelligence (AI), Machine Learning (ML), Deep Learning (DL), and Data Science, their tenets, and their current and upcoming developments. We also address future directions of AI, such as the transi-

tion to Artificial General Intelligence (AGI). Last but not least, we explore some of the ethical dilemmas posed by AI and data science advancements.

Chapter 3: The chapter begins with a review of the definition and taxonomy of Transfer Learning and Deep Transfer Learning (DTL). We also verify the WHY of transfer learning in deep learning: reducing the DL models' need for extensive labeled data and training costs. Then we investigate thirty-eight practical applications of Deep Transfer Learning approaches drawn from hundreds of publications using the systematic literature review methodology. Also, two thorough experimental analyses of applying DTL techniques are reviewed to find the best way of applying DTL in different scenarios and tasks. Moreover, the limitations of DTL approaches are investigated along with how this investigation shaped the concept of EXPANSE.

Chapter 4: We discuss limitations in the typical training process of deep learning models and state-of-the-art deep transfer learning techniques in this chapter. The primary limitations and gaps we investigate and try to address with our proposed methodology are:

1. Consideration of the quality of training samples in the process of training the model.
2. Dealing with catastrophic forgetting and overly biased pre-trained model.
3. The potential of deep continual learning in deep transfer learning.

Then we explain the details of our proposed methodology, EXPANSE. The first contribution of EXPANSE is a two-step training technique, i.e., learning fundamentals first and then adding uncertainties and complexities. Second, we expand the model vertically, adding new nodes on top of the pre-trained layers. Expanding the model vertically increases

a model's learning capacity with the goal of avoiding the overly biased issue and catastrophic forgetting dilemma in current DTL techniques.

Chapter 5: EXPANSE methodology is evaluated in this chapter by assessing its applicability and effectiveness. We first examine the two-step training method in a Deep Neural Network (DNN) model on the well-known handwritten digit recognition problem (MNIST). Then for the same problem and model, we apply model expansion. To verify the effectiveness of our proposed methods, we use the same model and data without utilizing EXPANSE to define the benchmark for comparison. Following our promising results, we apply EXPANSE to another well-known problem, Fashion MNIST. The evaluation analysis shows the effectiveness of EXPANSE and its potential in deep continual learning since the final model in our tests is still valid on the first step's training data in the transfer learning process.

Chapter 2

An Overview of Developments and Ethical Issues via Artificial Intelligence, Machine Learning, Deep Learning, and Data Science

2.1 Chapter Overview

Today, devices and applications powered by artificial intelligence (AI) include modes of transportation, home appliances, and mobile applications; in short, they are ubiquitous. Many people will have at least heard of AI and perhaps even subdivisions of AI such as Machine Learning (ML)

and Deep Learning (DL). Each of these represents an advanced tool of data science explored in this chapter. First, we briefly review the history of these developments in data science, tracing the history from the first mechanical computer in 1850 to the current state of DL in 2020. Each section overviews some basic definitions, tenets, and current and future developments. We discuss possible future directions of AI—including the transition to Artificial General Intelligence (AGI). Finally, we explore some of the ethical dilemmas posed by such advances and offer a call to data and social scientists to carefully consider the implications of these technologies.

2.2 Introduction

It is nearly impossible to move around modern society without encountering a device or application powered by artificial intelligence (AI). Weather forecasts, traffic signals, airplanes, factory lines, home appliances, and mobile applications are just a few examples of areas likely to encounter elements controlled by AI. Yet, there is even more happening under the surface with AI managing countless applications including internet traffic, gene-related research, and medical image and history analyzation. For most people today, deep learning, machine learning, and AI are all terms for which they are at least familiar.

Another body of work that most people will have heard of is data science and data analytics. Technological advances over the past few decades have transferred the possibility of generating, storing, sharing, and analyzing data to nearly everyone. With data now being a true commodity, some have said that data is the new oil or gold. For example, retailers are now able to gather information about their sales as well as their customers habits and preferences to greatly benefit both parties. Retailers can then use this information to intelligently predict customer shopping habits during other times of the year as well as control their supplies based on projected demands,

thus, not wasting time and money on unnecessary storage or creating shortages. This is just one example of the great advances made possible by data science and its varying applications. With advances such as autonomous driving now available, there is no telling where data science and AI might take us.

In this chapter, we briefly review the history of these developments in artificial general intelligence, artificial intelligence, machine learning, deep learning, and data science (see Figure 2.1), tracing the history from the first mechanical computer in 1850 to the current state of deep learning in 2022. We overview the many evolutions in AI and discuss possible future directions as well as some of the ethical dilemmas posed by such advances. Ultimately, our goal is to overview these processes for a lay audience who may not have intimate knowledge of AI and data science at large.

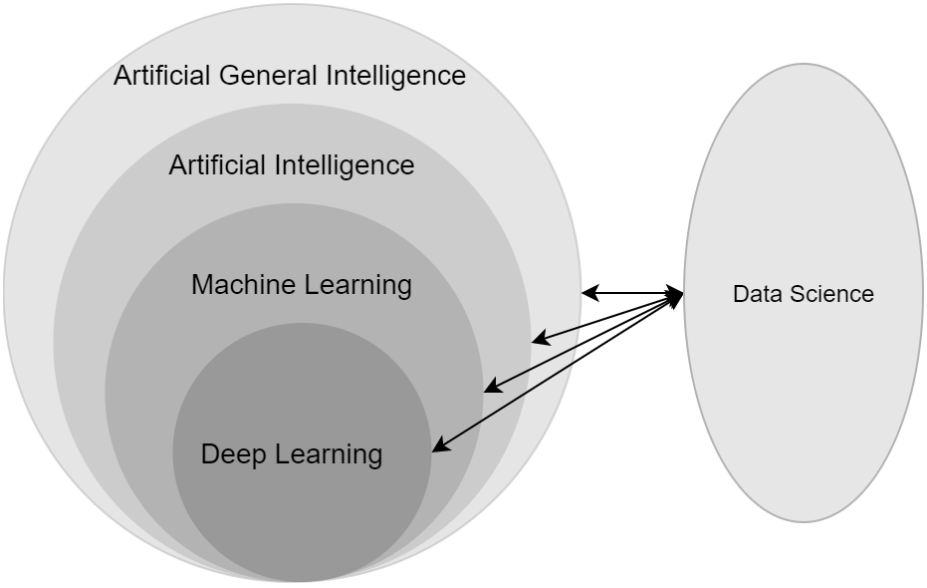

Figure 2.1: Domain hierarchy.

2.3 Artificial Intelligence (AI)

The first mechanical computer was invented in the 1850s by Charles Babbage [1]. In 1950, Alan Turing, renown for advancing the general-purpose programmable computer, asked the big question for the first time: "Can machines think?" [16]. Alan Turing proposed an operational test for machine intelligence. A machine "passes the test if a human interrogator, after posing some written questions, cannot tell whether the written responses come from a human or a machine." [16].

In 1956, the term "Artificial Intelligence" (AI) was used for the first time in a proposal for a summer research workshop at Dartmouth College in New Hampshire. The goal of AI was to "[make] a machine behave in ways that would be called intelligent if a human were so behaving" [17]. The aim of the workshop was to develop AI such that it might pass the Turing test.

To pass the Turing test the AI needs to [1]:

- Understand speech; *natural language processing (NLP)*
- Store the information and data; *knowledge representation*
- Use the stored data to draw conclusions; *automated reasoning/decision-making*
- Detect new patterns and adapt to new circumstances; *machine learning (ML)*

To fully pass the Turing test, two additional capabilities are needed [18][19]:

- Extract knowledge and/or comprehension from images or videos (e.g., face recognition); *computer vision*

- Mimic human physical behaviors corresponding with the senses to interact with the environment (e.g., touch, motor functioning); *physical interaction*

Overall, the AI could be divided into two categories of Artificial General Intelligence (AGI) or strong AI (actual thinking), and narrow or weak AI (simulated thinking) [1][20]. Some scholars argue that achieving AGI may be decades into the future, and that the emergence of AGI will bring with it an "intelligence explosion" leading to "profound changes in human civilization" [20]. Yet, the field of computer science has already begun to develop the narrower form of AI. In fact, the ability to have devices such as sensors and robots, and intelligent Decision Support Systems (DSS), such as the autocorrect software analyzing the words on this page are the result of already existing forms of AI [20]. These technologies have already evolved beyond what many people could have imagined, and yet the future of AGI has the potential to transform the experience of generations to come in ways we cannot yet predict.

Early (late 20th century) AI approaches were rule-based and focused on attending to all possible solutions for a specific and identifiable problem [21]. Some board games and various types of robots used in factory lines are just two examples of this type of rule-based AI. Going forward, decision making systems began to advance these types of approaches [22]. Specifically, decision making attended to the fact that real-life problems are rarely contained within such specific and rule-based features. Even board games must contend with players who make unpredictable decisions. Although decision making may follow some of the same patterns of rule-based prediction systems, it began to extend the bounds of these rules by accounting for uncertainty [23]. The Boltzmann machine research line during the 1990's through the early 2000's delivered a well-known example of this type of AI, which utilizes probability and statistics predicting behavior patterns in various settings [24][25][26]. Sum-Products Networks (SPNs) are another advance-

ment in AI that began to incorporate networks able to compete with deep learning models in many applications by taking into account the probability distributions of features [27].

The boom of digital storage development in the 1990s and 2000s – delivering cloud storage and advanced data collection methods – brought with it a new era of "big data." Big data refers to the vast amount of easily accessible consumer data including images, texts, audio, transactions, and human and environmental sensing data from electronic devices. This surge in available data required new methods for analyzing it, translating to Data Science (DS) and a new chapter in machine learning (ML).

2.4 Data Science

Data Science is divided into three main areas including collecting, storing, and analyzing data (structured or unstructured) [28]. Data collection methods were advanced through the spread of high-speed internet word wide – including less costly wireless connections – as well as increased variety of cheaper electronic connections and sensors, such as smartwatches, exercise trackers, and cameras [29]. Data storage was advanced through cloud storage, which further influenced big data collection by offering these services at a reduced cost and to an increasing proportion of the population.

Data analysis consists of two major components: preprocessing and processing. Preprocessing refers to various aspects of raw data management including unbalanced data, imputation techniques for missing data, detecting and addressing outliers, and data labeling procedures. Processing refers to extracting information and knowledge from preprocessed data to identify patterns, make predictions, and/or classify data [28]. One of the promising methodological categories for processing big data is Machine Learning (ML), a subdivision of AI.

2.5 Machine Learning (ML)

Machine learning (ML) is a subdivision of AI that consists of statistics, mathematics, and logical techniques to extract patterns (i.e., information) from a set of training data and apply the inferences to unseen data. Again, these recent advances in ML were made possible by the new era of big data and the vast advancement in computational capacity. Importantly, ML differs from other forms of AI in that it does not require extensive and complicated programming, but rather, has the ability to learn patterns and later apply them. Thus, ML does not need to consider every possible solution (i.e., be deterministic) and can manage noise and uncertainty [30].

Innovation in ML brought with its exponential advancement in earlier techniques – some of them developed before the 1970s – such as Linear Regressions, decision trees, Random forest, K-nearest neighbor (KNN), Support Vector Machine (SVM), Artificial Neural Networks (ANNs). For example, early ML ANN models for autonomous driving [31] and facial recognition [32] were developed in the 1980s but lacked access to the data and computation capacity needed to apply them [30].

Like any method, ML brings with it its own unique techniques and challenges. Common types of ML include supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Each of these is discussed in brief below, followed some of the challenges associates with ML such as overfitting and dealing with extraneous features.

Supervised learning refers to the use of an ML training data set that has been labeled, typically by humans, and the goal of which is to categorize or label the unseen data [33]. The process of categorization or labeling often occurs through classification or regression techniques, which

has value for making predictions – using regression – such as predicting stock market values or classifying objects in an image, such as identifying tumors in a medical x-ray.

Unsupervised learning refers to categorizing data by analyzing patterns and shared features without utilizing a pre-labeled training dataset [33]. In unsupervised learning, clustering is often used to detect patterns and anomalies, such as in grouping customers for marketing strategies or marking emails from unknown sources as "spam."

In addition, a smaller (i.e., limited) labeled ML dataset may be used to improve the categorization of a larger, unlabeled dataset. This is known as semi-supervised learning. Semi-supervised learning may be a more cost-effective option of labeling large datasets, in addition to allowing for greater accuracy by limiting human error [33]. For example, speech recognition errors may be reduced by 22% when human-labeled data are combined with machine-labeled data using semi-supervised learning [34].

Reinforcement learning (RL), operates using a reward-based system. Reinforcement learning attempts to select the best possible action that would maximize the final reward (or conversely, minimize the punishment), all while keeping track of these actions to improve the choice-selection of the following round. Thus, it is a trial-and-error process that works through the system's ability to learn improvement strategies and decisions through the success or failure of previous attempts. There are many different types of RL algorithms, each designed to address a specific problem [30]. Examples of RL applications include some types of board games (e.g., chess and Go), robots, and various elements of autonomous driving systems.

Although Machine Learning algorithms demonstrate immense accuracy in identifying training dataset patterns, a common problem in these models is overfitting the data [35], [36]. Overfitting occurs when the ML network has been trained using all labeled (i.e., training) data and

cannot deal with the noise (i.e., uncertainty) in the unseen data. It also occurs when patterns observed in the limited training data are not accurate of the existing patterns in the larger data. Overfitting may occur when using unbalanced or biased datasets, indicating that the training set does not include all possible samples within the domain[30], [35], [36].

There are several ways of correcting for the risk of overfitting. One of these is to divide the training dataset into two parts: training and validation [30][36]. The size of the validation set will depend on the size of the overall training set, but typically ranges from about 10-30% of the full set. The validation set is not used for training purposes, but is instead verified against the final dataset to ensure accuracy. In this procedure, cross-validation is used to correct for the risk of selecting a biased validation set [37]. For example, a 10-fold cross-validation procedure would involve dividing a training set into 10 separate sets and then training the ML model 10 times using only nine of those sets each time. The final model would then be validated against the remaining set (1/10th of the original), with the accuracy being equal to the average of the 10 validation runs.

Another challenge that may arise in ML models is the issue of extraneous features, such as the vast number of potentially uncorrelated features present in some big data sets. In many cases, not all of the features present in a dataset will be related to the objective of the ML model and, thus, are not useful. For example, to predict the seasonal sales of an online store, customers' employment status and income may be related to the outcome, but their specific job title may not be. There are several known processes for responding to unrelated features in a dataset including feature selection, combination, and extraction [38]. These are performed through techniques like correlation analyzation, principal component analyzation (PCA), and dimensionality reduction techniques. These techniques work mainly by validating the correlation of each feature to the target [38].

Machine Learning techniques and data sets can be categorized into two groups: linear and non-linear. A linear data pattern is the simplest data pattern and can be categorized using a linear function to perform regression or classification. Many algorithms had been developed to fit linear models such as linear regression, logistic regression, classification and regression trees, K-nearest neighbors, and support vector machine [30].

Non-linear functions are those that cannot be classified using linear methods. Like other models of data analysis and management, non-linear data associations may pose additional challenges to ML [30]. The non-linear problem in ML is known as the XOR (i.e., "exclusive or") problem, which refers to a mixed pattern of data that cannot be categorized using linear functions, Figure 2.2 [39].

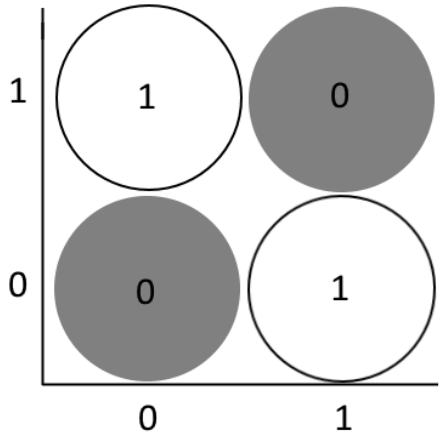

Figure 2.2: Not linearly separable pattern, known as XOR problem.

Although many algorithms have been developed to manage linear data (as mentioned above), the non-linear nature of many data sets remained a challenge for ML. For example, the decision trees, k-nearest neighbors, and support vector machine mentioned above are functions that can

manage some non-linear data problems; yet, they do this imperfectly, and issues remain. Artificial Neural Networks (described in the next section) began to address these issues [30].

2.6 Artificial Neural Network (ANN)

In 1958, the first artificial neuron was introduced—attempting to mimic the neural pathways of the human brain. Named Perceptron, it used a sigmoid function and performed linear functions with great success [40]. To advance this then new technology, several Perceptrons were later clustered into a layer, allowing for linear patterns to be detected through the use of input data connecting into the Perceptron layer. Training happens by feedforwarding the data while backpropagating the labels to tune the weights of each node. Thus, the first artificial neural network (ANN) was born [41]. Perceptron remained at the height of ANN mechanisms until 1969, when rigorous reviews demonstrated its shortcomings—namely, that Perceptron could not address the issue of nonlinearity; it had hit a dead-end [42].

By the 1980s, scientists again attempted to address the issue of nonlinearity (i.e., the XOR problem) by using hidden layer(s) of Perceptron, known as Multilayer Perceptron (MLP). MLP is a type of ANN consisting of one or more layers of varying nodes—the network architecture (see Figure 2.3) [41]. Using an activation function, such as sigmoid, on the front end of the nodes, again combined with backpropagation techniques, allowed for increasingly advanced classification and regression models—including those for nonlinear patterns [41]. These advances greatly improved the accuracy of some of the advanced technologies we enjoy today, such as autonomous driving and facial recognition.

The early ANN designs were fully connected, with each node tied to the next, and each connection having a weight. Each node contained an activation function and uses the value of

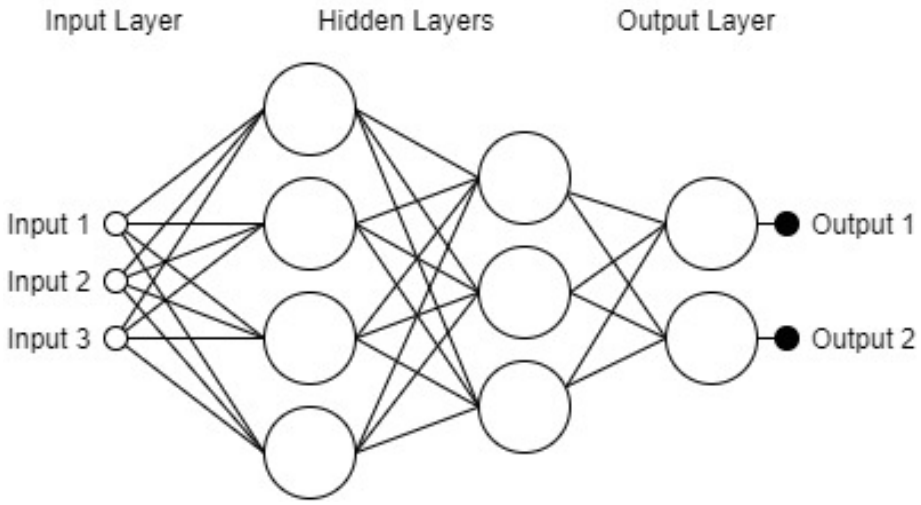

Figure 2.3: An example of MLP network architecture.

prior nodes multiplied by the weight of the connection to calculate the next node in a recursive loop. The simultaneous backpropagation by means of the training dataset serves to update and fine-tune the node weights and thresholds. Similar to the reinforcement learning process described above, cost/loss functions are used as additional metrics by which to measure the compatibility between the training data (i.e., ground truth) and the network predictions [41].

Despite vast advances in theorizing, the ANNs (MLP) of the 1980s faced several challenges. Specifically, the limited number of available nodes in each MLP layer, combined with the limited number of layers, produced a heavy burden for the computers of the day. In short, advanced theorizing was limited to the computational capacity of the 1980's machines. However, by the year 2000, significant advances were made in computational capacity. These advances, paired with the ability to replace the nodes' sigmoid activation function with more efficient functions such as sign, linear, tanh [43][44], and more recently ReLU and leaky-ReLU [45], allowed for the

creation of a larger network of nodes, including more hidden layers. This led to the creation and advancement of deep neural networks (DNN), also known as deep learning (DL).

2.7 Deep Learning (DL)

As mentioned, the vast improvements in the computational capacity of the 2000s helped shape the development of deep neural networks (DNN) or deep learning (DL). Another shaping factor in the development of DL was the arrival of big data sets, which offered the opportunity to improve the training process and thus the performance of DNN.

Similar to ANNs, the learning in DNNs occurs through the optimization of the weights throughout the entire network. One of the well-known algorithms for handling this type of optimization problem is Stochastic Gradient Descent (SGD) [46]. There are several other methods based on the SGD algorithm such as Momentum, Nesterov Momentum, and Adam [46]. Each of these methods works by tracing the error surface of the error calculation function (known as loss function) with the goal of finding the global minima, as shown in Figure 2.4 [46] [47]. The loss function is based on the adjustments of the weights of each of the connections in the network [46].

Other parameters that need to be taken into consideration in order to maximize DNNs' accuracy include data preprocessing, hyperparameter adjusting such as learning rate adjustments, weight initialization, initializing biases, and batch normalization [48][49].

Several modifications of DNNs have vastly improved the implementation of these models. The modifications aim to reduce the models' generalization error by regularizing the weights. There are several methods to do such regularizations including considering the noise robustness, stop learning point (i.e., early stopping), parameter sharing, and dropout [48][49]. The following

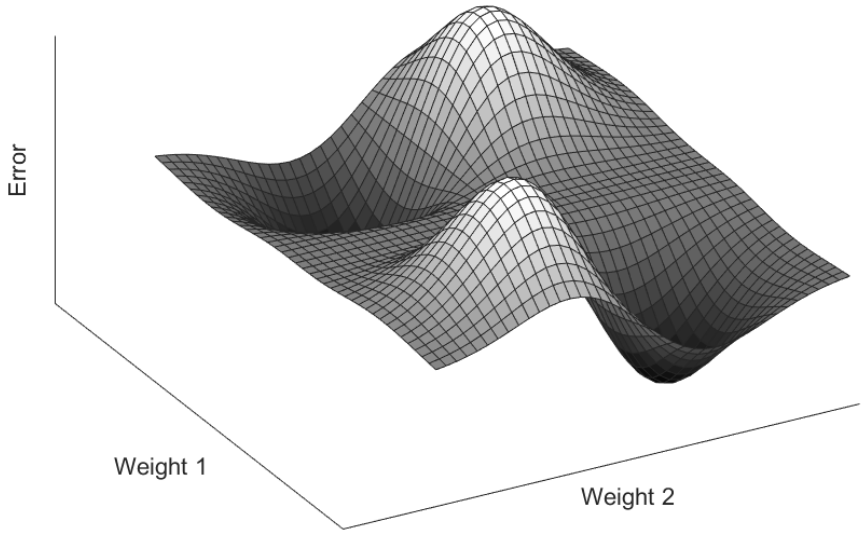

Figure 2.4: An example of error surface in an ANN/DNN.

section overviews some of the main events and advancements in determining the current state of DLs.

In 2007, Fei Fei Li and colleagues introduced ImageNet, the largest database of labeled images with over 14 million images categorized into nearly 22,000 indexed synsets (categories) as of 2020 [image-net.org]. These images can be used for technologies such as object location, detection, and classification in videos and other image-related media [image-net.org]. Since 2010, ImageNet has led an annual challenge – the ImageNet Large Scale Visual Recognition Challenge (ILSVRC). The challenge brings bright minds from around the globe together to explore new ideas in DL by allowing them to use a large collection of image-data they would not otherwise have access to. This challenge has brought great success in minimizing error as demonstrated in Figure 2.5. Remarkably, the classification error of 28% in 2010 was reduced to less than 3% by 2017 [50].

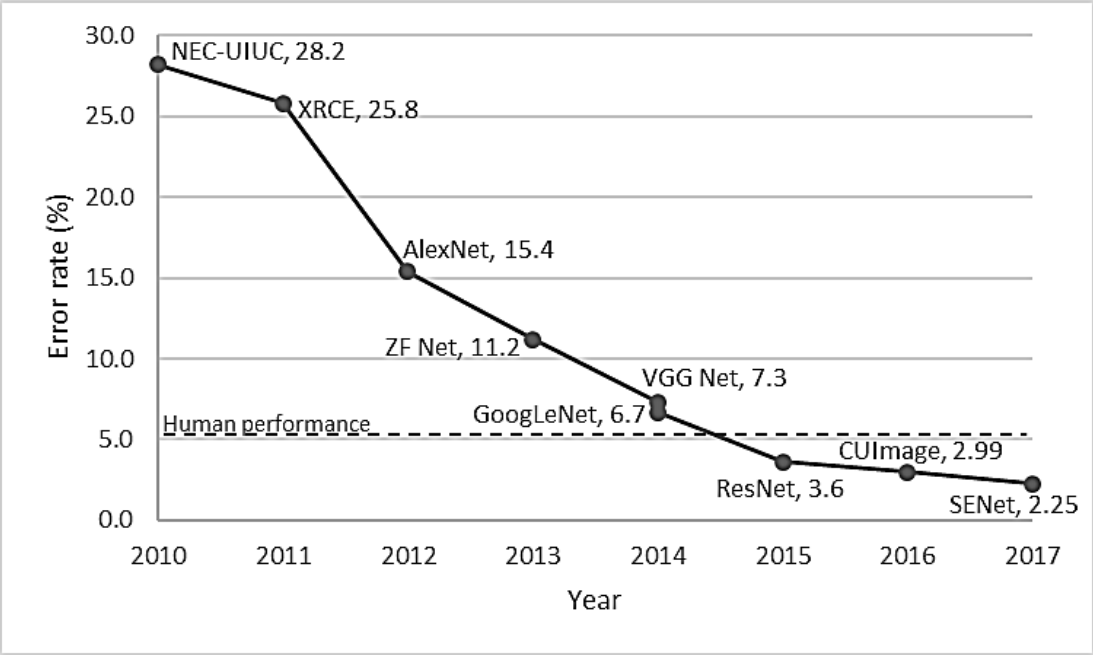

Figure 2.5: ILSVRC winners.

By 2011, the Convolutional Neural Network (CNN) was beginning to grow in popularity. CNN was able to outperform humans in recognition of traffic signs with an accuracy of 99.46% (compared to humans at 99.22%) [51]. First introduced in 1997, CNN was inspired by the visual cortex of animals and attempted to regularize input data to find hierarchical patterns within image data—called self-organized map [52]. By 2020, nearly all DLs utilized CNN layer(s) for visual-related tasks. Going back to the ILSVRC, CNN was utilized by the majority of champions, but it has been used for many other applications as well [50]. Interesting research conducted in 2015 demonstrates how a DNN network functions using CNN layers (i.e., https://youtu.be/AgkfIQ4IGaM) [53].

Another network architecture from the year 1997, Long Short-Term Memory (LSTM) [54], also saw vast improvements in the 2010 decade [55]. LSTM, also known as Recurrent Neural Network (RNN), is a modified neural network that utilized feedback connections. LSTM allowed for the subsequent advanced in DL such as speech and handwriting recognition applications, as well as anomaly detection in data series (e.g., network traffic) [55].

In 2014, Ian Goodfellow and colleagues invented the Generative Adversarial Network (GAN) [56]. GAN is comprised of two neural networks competing against each other; the first is a generative network which generates new data while the second is a discriminative network that evaluates the generated data. This advanced network can generate new data based on the characteristics of the inputted training data. For example, if the training data were to be of a human face, the network would generate a new face, looking entirely human but never having previously existed [57]. A vast amount of applications benefits from GANs, such as imaginary fashion models and scientific simulations [57]. Despite their many advances, GANs raise some concerns, specifically regarding the production of falsified voice or video records [58].

By 2016, Google announced the Tensor Processing Unit (TPU), followed by the Google TensorFlow framework of open source libraries [59]. TensorFlow touts well-tailored hardware and software to be used for neural network computations and applications [59]. Using this technology, Google's DeepMind AlphaGo defeated Go champion in 2016 by combining DL and RL in a new mechanism named Deep Reinforcement Learning [60].

Another contemporary topic in ML that was also initiated in the 1990s is transfer learning or domain adaptation, first published by Lorien Pratt [61]. Transfer learning works by using knowledge garnered through the ML model during the training phase and literally transferring the learning to another task in a similar domain [62]. For example, a DL model trained to classify

flowers can be used to also classify leaves by modifying the trained model using a transfer learning technique (e.g., fine-tuning some of the layers). Since 2014, transfer learning has been used to adapt deep learning models, such as in domains like medical imaging. This is known as deep transfer learning and has been used to reduce the often-long training time as well as to handle the small training samples of some deep learning [62]. Progressive learning introduced by Google's DeepMind Project in 2016 is another specific type of deep transfer learning that is attempting to build on previous, related knowledge, similar to human learning capabilities [63].

In summary, this overview summarized a few notable types of DL that are on the rise. It is important to note that the aforementioned advancements in DL are vast topics in and of themselves, each carrying with them a research line with hundreds or even thousands of relevant articles that could not be overviewed here. In addition, there are many more DL advances not discussed here, including the autoencoders used for image segmentation models such as U-Net and new types of data compressors [64], among many others.

2.8 Discussion

As mentioned, the era of big data, spurred by drastic advancements in computational capacity, has brought a new chapter to machine learning (ML) and artificial intelligence (AI) since the turn of the century. In the past decade alone, the movement toward artificial general intelligence (AGI) has grown exponentially, and there is no telling where it might take us.

A common example of the great innovations of AGI is IBM's Watson, first introduced in 2011. Watson is a natural language processing (NLP) platform, whose architecture benefits from a variety of developments in AI and ML [65]. In 2011, Watson defeated the champions of the

popular quiz show Jeopardy—a feat spurred by its ability to "process 500 gigabytes, the equivalent of a million books, per second" [66].

The use of even narrow AIs to mimic human cognition is opening the pathway to AGI, and AGI is a force that future humans will have to contend with. The competition is likely to be intense given that computer programs do not suffer from fatigue, boredom, or other common human ailments—and impediments to work and/or output. For example, Google's well-known program, AlphaGo, managed to train itself in one night to rise from an amateur to a champion player by the next morning [60].

There are countless other examples of the ways in which AGI is looming closer. Present-day Artificial Neural Networks (ANN) are already a simple mimic of human brain cells, and Convolutional Neural Networks (CNN) mime the human visual cortex. Generative Adversarial Networks (GAN) work like the human imagination—generating new data from observed data—which can be used for better understanding facts by imaging-related data. All this is done without needing to access all or even a vast amount of data. Long-Short Term Memory (LSTM) works similarly to the human memory and is able to solve problems related to sequential data analyzing. Transfer learning, followed by progressive learning, is attempting to mimic human skill-learning abilities, a task that is endless for human beings. However, human beings must rely on previous knowledge and skills—oftentimes garnered over a lifetime—that AI programs can learn in a matter of hours. All of this evidence suggests that AGI is close to becoming reality, and the implications of this have yet to be explored.

2.9 Ethics in AI and Data Science

With any scientific advance—particularly one that travels so quickly—ethical issues and considerations are unavoidable. Recent years have seen an increase in considerations of the potential ethical pitfalls of AI and the use of personal data raised by data scientists and other scholars (including social scientists, historians, and others) [67][68][69]. Unfortunately, the ethical consequences of many advances are difficult to assess in real-time. For example, although it is relatively obvious to see the issues with falsifying evidence through GANs [58], the impact of wrongly classifying a disease through X-ray images is more difficult to project, not to mention the social implications of these advanced changes. This section discusses some primary areas of concern in the ethics debate surrounding AI and offers some additional points for consideration.

One primary area of ethics currently involves data privacy surrounding sensitive data and personal information such as credit card transactions and medical data. Issues related to data privacy are complicated by the need for access to personal information in order to move many fields forward. For example, not using medical information means that patients miss out on the opportunity to have their diagnoses made by more accurate AI programs. On the other hand, there are also consequences to this data being made available. Doctors risk being sued if later AI advances point to something they missed, and patients are at risk of having their private information shared elsewhere. Like most ethical issues, it is imperative to consider both sides of this debate and to seek solutions that maximize benefits while limiting risks. Data anonymization mechanisms begin to address these issues by making data unidentifiable, which allow for the positive usage of private information without risking patient or physician privacy [70].

A second ethical implication includes the social impact of human job loss if AI automates such jobs. For example, the rise of autonomous driving semis and other transport vehicles has the potential to contribute to the unemployment of a large proportion of middle-class workers in the United States. This process is similar to the transition of farming to factory jobs across Europe and other parts of the world during the industrial era as well as the subsequent automation of factory lines. During these times, many workers lost their jobs, thereby moving from middle-class into poverty. Although these changes are unavoidable, the social impact on families and communities must be considered. However, existing data and insight from the industrial and automation booms can help data scientists and social science researchers better predict and prepare for the implications of AI-automation for employment. Planning for these potential consequences may help to ease the transition for society and future generations.

A final ethical implication worth noting here is the broader impact on society. Such impacts are often difficult to observe in real time. One poignant example of the effects of AI is political polarization. With the rise of social media and worldwide connectivity via cell phones, tablets, smartwatches, and other devices, the implications of AI automating what information people have access to is more evident than ever. Some have suggested that the targeted marketing and information brought by AI has contributed to political and social polarization, with many people having access only to the information they already agree with. Opinions are constantly being validated, and the ability to be objective in any topic is becoming limited. This process has also occurred with consumer branding, as AI approaches target consumers with ads for products they are more likely to buy based on their previous purchasing behaviors. In fact, it has been said that these mechanisms know people better than they know themselves. The implications of the human mind becoming inundated with certain types of data remain to be seen, and the ethical

considerations have yet to be examined. However, such implications must be on our radar as they have the potential to change society for generations.

Ethical dilemmas are just that: dilemmas. The vast majority are not easily solvable or even identifiable. However, their elusiveness cannot be the reason that data scientists fail to consider these issues—potential or currently a reality. Rather, it is the responsibility of the data scientist community to partner with other disciplines (e.g., social and behavioral sciences) to consider the effects of their creations on society, no matter how far into the future they may reach.

Chapter 3

Deep Transfer Learning and Recent Advancements (Literature Review)

3.1 Chapter Overview

Deep learning has been the answer to many machine learning problems during the past two decades. However, it comes with two major constraints: dependency on extensive labeled data and training costs. Transfer learning in deep learning, known as Deep Transfer Learning (DTL), attempts to reduce such dependency and costs by reusing an obtained knowledge from a source data/task in training on a target data/task. Most applied DTL techniques are network/model-based approaches. These methods reduce the dependency of deep learning models on extensive training data and drastically decrease training costs. As a result, researchers detected Covid-19

infection on chest X-Rays with high accuracy at the beginning of the pandemic with minimal data using DTL techniques. Also, the training cost reduction makes DTL viable on edge devices with limited resources. Like any new advancement, DTL methods have their own limitations, and a successful transfer depends on some adjustments for different scenarios. In this chapter, we review the definition and taxonomy of deep transfer learning and well-known methods. Then we investigate the DTL approaches by reviewing recent applied DTL techniques in the past five years. Further, we review some experimental analyses of DTLs to learn the best practice for applying DTL in different scenarios. Moreover, the limitations of DTLs (catastrophic forgetting dilemma and overly biased pre-trained models) are discussed, along with possible solutions and research trends.

3.2 Introduction

In recent years, Deep Learning (DL) has successfully addressed a number of challenging and interesting applications; in particular, problems that involved non-linearity of datasets. Recent advancements in deep learning methods deliver various usages and applications in extremely different areas such as image processing, natural language processing (NLP), numerical data analysis and predictions, and voice recognition. However, deep learning comes with restrictions, such as expensive training processes (time and processing) and the requirement of extensive training data (labeled data) [2].

Since the start of the Machine Learning (ML) era, transfer learning has been a neat exploration for scientists. Before the rise of deep learning models, transfer learning was known as domain adaptation and focused on homogeneous data sets and how to relate such sets to each other because of the nature of ML algorithms [3], [71]. Traditional ML models have less dependency

on dataset size, and usually, their training is less costly than deep learning models since they have been mostly designed for linear problems. Therefore, the motivation for using transfer learning in deep learning is higher than ever in the AI (Artificial Intelligence) and ML fields since it can address the two restraints of extensive training data and training costs.

Recent transfer learning methods on deep learning aim to reduce training process time and cost, and the necessity of extensive training datasets which can be hard to harvest in some areas such as medical images. Moreover, a pre-trained model for a specific job can be run on a simple edge device like a cellphone with limited processing capacity and limited training time [72]. Also, developments in DTL are opening the door to more intuitive and sophisticated AI systems since it considers learning a continuous task. A great example of this idea is Google's deep mind project and advancements such as progressive learning [7]. All this is bringing DTL to the forefront of research in artificial intelligence and machine learning.

In this chapter, first, the definition of DTL is reviewed, followed by the taxonomy of DTL. Then, selected recent practical studies of DTL are listed, categorized, and summarized. Moreover, two experimental evaluations of DTL and their conclusions are reviewed. Last but not least, we discuss the limitations of today's DTL techniques and possible ways to tackle them.

3.3 Deep Learning

Deep learning (DL) or deep neural network (DNN) is a machine learning subcategory, which can deal with nonlinear datasets. DNNs consist of layers of stacked nodes, with activation function and associated weights, (fully/partially) connected and usually trained (weight adjustments) by back-propagation and optimization algorithms. During the past two decades, DNNs were developed rapidly and are used in many aspects of our daily lives today. For instance, Convolutional

Neural Network (CNN) layers have improved deep learning models for visual-related tasks since 2011, and as of today, most DLs use CNN layers [2].

3.4 Deep Transfer Learning (DTL)

Deep transfer learning is about using the obtained knowledge from another task and dataset (even one not strongly related to the source task or dataset) to reduce learning costs. In many ML problems arranging a large amount of labeled data is impossible, which is mandatory for most DL models. For instance, at the beginning of the Covid-19 pandemic or even a year into it, providing enough chest X-Ray labeled data for training a deep learning model was still challenging, while using deep transfer learning, the AI achieved detecting the disease with very high accuracy with a limited training set [73], [74]. Another application is applying machine learning on edge devices such as phones for variant tasks by taking advantage of deep transfer learning to reduce the need for processing power.

An untrained DL uses a random initializing weight for nodes, and during the expensive training process, those weights adjust to the most optimized values by applying an optimization algorithm for a specific task (dataset). Remarkably, [75] proved that initializing those weights based on a trained network with even a very distant dataset improves training performance compared to the random initialization.

Deep transfer learning differs from semi-supervised learning since, in DTL, the source and target datasets can have a different distribution and just be related to each other, while in semi-supervised learning, the source and target data are from the same dataset, only the target set does not have the labels [71]. DTL is also not the same as Multiview learning since Multiview learning uses two or more distinct datasets to improve the quality of one task, e.g., video datasets can

be separated into image and audio datasets [71]. Last but not least, DTL differs from Multitask learning despite many shared similarities. The most fundamental difference is that in Multitask learning, the tasks use interconnections to boost each other, and knowledge transfer happens concurrently between related tasks. In contrast in DTL, the target domain is the focus, and the knowledge has already been obtained for target data from source data, and they do not need to be related or function simultaneously [71].

3.5 From Transfer Learning to Deep Transfer Learning, Taxonomy

It is possible to categorize Deep Transfer Learnings (DTLs) in different ways by various criteria, similar to Transfer Learnings. DTLs can be divided into two categories of homogeneous and heterogenous based on the homogeneity of source and target data [71]. However, this categorization can be done differently because it is subjective and relative. For example, a dataset of X-Ray photos can be considered heterogeneous to a dataset of tree species photos when the comparison domain is limited to only image data. In contrast, it can be considered homogeneous to the same tree species photo dataset when the domain consists of audio and text datasets.

Also, DTLs can be categorized into three groups based on label-setting aspects: (i) transductive, (ii) inductive, and (iii) unsupervised [71]. Briefly, transductive is when only the source data is labeled; if both source and target data are labeled it is inductive; if none of the data are labeled it is unsupervised deep transfer learning [71].

[71] and [76] mention and define another categorization of DTLs through the aspect of applied approaches. They similarly categorized DTLs into four groups of: (i) instance-based, (ii)

feature-based / mapping-based, (iii) parameter-based / network-based, and (iv) relational-based / adversarial-based approaches. Instance-based transfer learning approaches are based on using selected parts of instances (or all) in source data and applying different weighting strategies to be used with target data. Feature-based approaches map instances (or some features) from both source and target data into more homogeneous data. Further, the [71] survey divides the feature-based category into asymmetric and symmetric feature-based transfer learning subcategories. "Asymmetric approaches transform the source features to match the target ones. In contrast, symmetric approaches attempt to find a common latent feature space and then transform both the source and the target features into a new feature representation." [71] The network-based (parameter-based) methods are about using the obtained knowledge in the model (network) with different combinations of pre-trained layers: freezing some and/or finetuning some and/or adding some fresh layers. Relational/adversarial-based approaches focus on extracting transferable features from both source and target data either using the logical relationship or rules learned in the source domain or by applying methods inspired by generative adversarial networks (GAN) [71], [76]. Figure 3.1 shows the taxonomy of the above-mentioned categories [71].

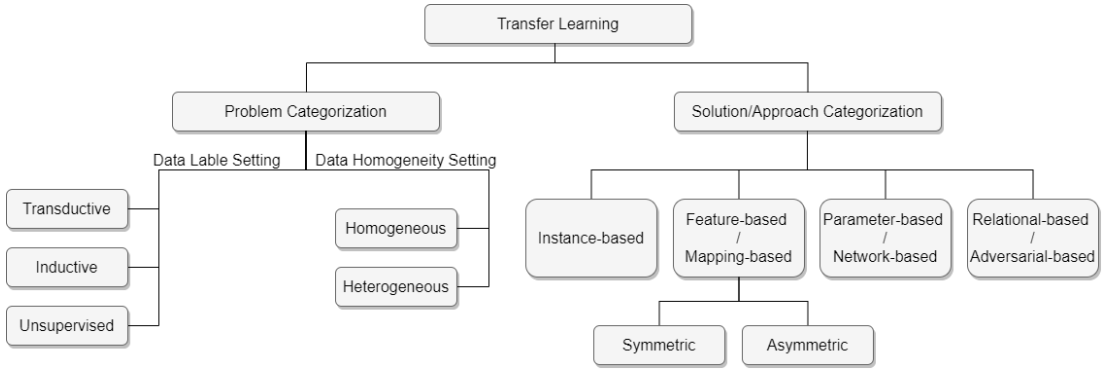

Figure 3.1: Taxonomy of Transfer Learning which is extendable to Deep Transfer Learning as well.

Other than the network-based and adversarial-based approaches, all other categories have been explored deeply during the last couple of decades for different ML techniques known as domain adaptation or transfer learning [3], [71]. However, most of those techniques are still applicable to deep transfer learning (DTL) as well. Network-based (parameter-based) approaches are the most applied techniques in DTL since they can tackle the domain adaptation between source and target data by adjusting the network (model). In other words, deep transfer learning is mainly focused on network-based approaches. Remarkably, network-based approaches in deep learning models can even tackle the adaptation of a very distant source and target data [71], [76].

In deep transfer learning (DTL), different techniques are applied for network-based approaches, although generally, they are combinations of pre-training, freezing, finetuning, and/or adding a fresh layer(s). A deep learning network (DL model) trained on source data is called a pre-trained model consisting of pre-trained layers. Freezing and finetuning are techniques using some or all layers of pre-trained models to train the model on target data. Freezing some layers means the parameters/weights will not change and are constant values for frozen layers from a pre-trained model. finetune means the parameters/weights are initialized with the pre-trained values instead of random initialization for the whole network or some selected layers. Another recent DTL technique is based on freezing a pre-trained model and adding new layers to that model for training on target data; Google's deep mind project introduces this technique in 2016 as Progressive Learning / progressive neural networks (PNNs) [7], [77].

The concept of progressive learning mimics human skill learning, which is adding a new skill on top of previously learned skills as a foundation to learn a new one. E.g., a child learns how to run after learning to crawl and walk and using all the skills obtained in the process. Similarly, PNNs prevent catastrophic forgetting in DTL versus finetuning techniques by freezing the whole

pre-trained model and learning (adjusting to) the new task by training the newly added layers on top of previously trained layers [7], [77].

In deep learning models, usually, the earlier layers do the feature extraction at a high level of detail, further layers towards the end extract the information and conceptualize the given data, and lateral layers do the classifications or predictions. For instance, in the image-related model, the earlier layers of CNN extract the edges, corners, and tiny patches of a given image. Further layers put those details together to detect objects or faces, and the lateral layers, usually fully connected layers, do the classification [14]. Given this process, the most effective and efficient approach for DTL, to our knowledge, is to freeze the earlier and middle layers from a related pre-trained model and finetune the lateral layers for the new task/dataset [78]. Similarly, the new layers are added to the last part of a pre-trained model in progressive learning.

Nonetheless, some other research in this area use combinational and sophisticated methods to tackle transfer learning in deep learning like ensembled networks, weighting strategies, etc. [71]. However, to our knowledge, the search for recent advancements in DTL for practical tasks ends up with methods based on mostly the network-based and limited number of adversarial-based approaches.

3.6 Review of Recent Advancements in DTL

We limited our selection to the last five years of published studies on deep transfer learning for various tasks and data types. Table 1 shows the list of selected works from hundreds of reviewed literature sorted by their DTL approaches. We used the systematic literature review (SLR) technique [79] for the process of finding and selecting these thirty-eight publications. The inclusion criteria that we used for our selection process are as follows: a) published in the past five years, b)

reproducible (detailed implementation and models), c) applied to practical ML problems, and d) generalizable. We found that all reviewed studies mostly fall into three categories of network-based approaches and some into the adversarial-based approach, which are explained in the previous section. We name these approaches as (i) Finetuning: finetuning a pre-trained model on target data; (ii) Freezing CNN layers: the earlier CNN layers are frozen, and only the lateral fully connected layers are finetuned; (iii) Progressive learning: some or all layers of a pre-trained model are selected and used frozen, and some fresh layers will be added to the model to be trained on target data; and (iv) Adversarial-based: extracting transferable features from both source and target data using adversarial or relational methods, Figure 3.2.

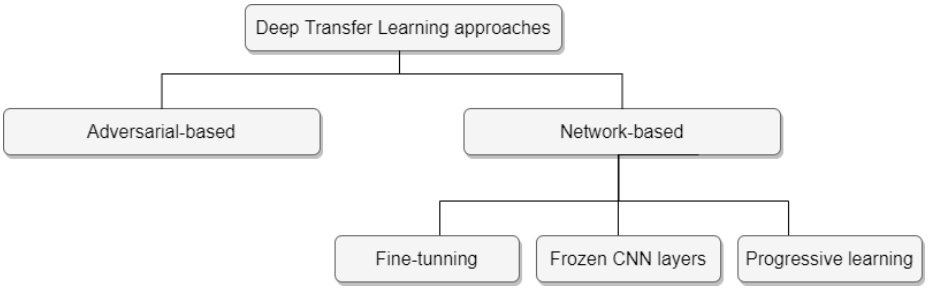

Figure 3.2: Most common Deep Transfer Learning approaches.

The most common DTL method is using a trained model on a highly related dataset to target data and finetune it on target data (finetuning). The simplicity of applying this technique makes it the most popular DTL method in our selection; 21 of 38 selected works have used this method. This method can improve training on target data in various ways, such as reducing training costs and tackling the need for an extensive target dataset. However, it is still prone to catastrophic forgetting. Needless to say, it is a very effective DTL method for many tasks and datasets in various fields such as medical, mechanics, art, physics, security, etc. Also, it has been applied for both image datasets and tabular (numerical) datasets as listed in Table 3.1.

5

Table 3.1: List of selected recent deep transfer learning (DTL) publications.

Begin of Table								
Ref.	**Year**	**Title**	**Data Type**	**Time Series**	**Approach**	**CNN**	**Known Models Used**	**Dataset Field**
[80]	2022	UAV swarm based radar signal sorting via multi-source data fusion: A deep transfer learning framework	Image	no	Finetuning	yes	Yolo, Faster-RCNN, Cascade-RCNN	Radar image
[81]	2022	Classification of analyzable metaphase images using transfer learning and fine tuning	Image	no	Finetuning	yes	VGG16, Inception V3	Medical image
[82]	2021	Multiclassification of Endoscopic Colonoscopy Images Based on Deep Transfer Learning	Image	no	Finetuning	yes	AlexNet, VGG, Res-Net	Medical Image
[83]	2021	MCFT-CNN: Malware classification with fine-tune convolution neural networks using traditional and transfer learning in Internet of Things	Image	no	Finetuning	yes	Res-Net50	Malware classification
[84]	2021	Facial Emotion Recognition Using Transfer Learning in the Deep CNN	Image	no	Finetuning	yes	VGGs, Res-Nets, Inception-v3, DenseNet-161	Facial emotion recognition (FER)

Continuation of Table 3.1

Ref.	Year	Title	Data Type	Time Series	Approach	CNN	Known Models Used	Dataset Field
[73]	2020	Automated Deep Transfer Learning-Based Approach for Detection of COVID-19 Infection in Chest X-rays	Image	no	Finetuning	yes	Inception-Xception	Medical image
[74]	2020	Classification of the COVID-19 infected patients using DenseNet201 based deep transfer learning	Image	no	Finetuning	yes	ImageNet, Dense-Net	Medical image
[85]	2019	Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning	Tabular/ bigdata	yes	Finetuning	no	none	Quantum mechanics
[86]	2019	Application of deep transfer learning for automated brain abnormality classification using MR images	Image	no	Finetuning	yes	Res-Net	Medical image
[87]	2019	An adaptive deep transfer learning method for bearing fault diagnosis	Tabular/ bigdata	yes	Finetuning	no	LSTM RNN	Mechanic
[88]	2019	Online detection for bearing incipient fault based on deep transfer learning	Image	yes	Finetuning	yes	VGG-16	Mechanic
[89]	2019	Towards More Accurate Automatic Sleep Staging via Deep Transfer Learning	Tabular/ bigdata	yes	Finetuning	yes	none	Medical data
[90]	2019	Deep Transfer Learning for Multiple Class Novelty Detection	Image	no	Finetuning	yes	Alex-Net, VGG-Net	Vision

Continuation of Table 3.1

Ref.	Year	Title	Data Type	Time Series	Approach	CNN	Known Models Used	Dataset Field
[91]	2019	A Digital-Twin-Assisted Fault Diagnosis Using Deep Transfer Learning	Tabular/ bigdata	no	Finetuning	no	none	Mechanic
[92]	2019	Learning to Discover Novel Visual Categories via Deep Transfer Clustering	Image	no	Finetuning	yes	none	Vision
[93]	2018	Deep Transfer Learning for Person Re-identification	Image	no	Finetuning	yes	none	Identification / security
[94]	2018	Deep Transfer Learning for Art Classification Problems	Image	no	Finetuning	yes	none	Art
[95]	2018	Classification and unsupervised clustering of LIGO data with Deep Transfer Learning	Image	no	Finetuning	yes	none	Physics / Astrophysics
[96]	2018	Empirical Study and Improvement on Deep Transfer Learning for Human Activity Recognition	Tabular/ bigdata	yes	Finetuning	yes	none	Human Activity Recognition
[97]	2018	Automatic ICD-9 coding via deep transfer learning	Tabular/ bigdata	no	Finetuning	yes	none	Medical
[98]	2017	Video-based emotion recognition in the wild using deep transfer learning and score fusion	video (audio & visual)	yes	Finetuning	yes	VGG-Face	Human science / psychology

Continuation of Table 3.1

Ref.	Year	Title	Data Type	Time Series	Approach	CNN	Known Models Used	Dataset Field
[99]	2022	Deep transfer learning-based visual classification of pressure injuries stages	Image	no	Freezing CNN layers	yes	Dense-Net 121, Inception V3, MobilNet V2, Res-Nets, VGG16	Medical image
[100]	2021	Deep Transfer Learning for WiFi Localization	Tabular/ bigdata	no	Freezing CNN layers	yes	none	WiFi Localization
[101]	2020	Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images	Image	no	Freezing CNN layers	yes	Res-Net, Dense-Net	Medical image
[102]	2019	Deep Transfer Learning for Signal Detection in Ambient Backscatter Communications	Tabular/ bigdata	no	Freezing CNN layers	yes	none	Telecommunication
[103]	2019	Brain tumor classification using deep CNN features via transfer learning	Image	no	Freezing CNN layers	yes	Google-Net	Medical image
[104]	2018	Comparison of Deep Transfer Learning Strategies for Digital Pathology	Image	no	Freezing CNN layers	yes	none	Medical image
[105]	2018	Deep transfer learning for military object recognition under small training set condition	Image	no	Freezing CNN layers	yes	none	Military

Continuation of Table 3.1

Ref.	Year	Title	Data Type	Time Series	Approach	CNN	Known Models Used	Dataset Field
[106]	2018	Deep Transfer Learning for Image-Based Structural Damage Recognition	Image	no	Freezing CNN layers	yes	VGG-Net	Civil engineering
[107]	2017	Deep Transfer Learning for Modality Classification of Medical Images	Image	no	Freezing CNN layers	yes	VGG-Net, Res-Net	Medical image
[108]	2017	Folding Membrane Proteins by Deep Transfer Learning	Tabular/ bigdata	no	Freezing CNN layers	yes	Res-Net	Chemistry
[109]	2021	Progressive Transfer Learning Approach for Identifying the Leaf Type by Optimizing Network Parameters	Image	no	Progressive learning	yes	Res-Net50	Plant science
[110]	2020	An Evaluation of Progressive Neural Networks for Transfer Learning in Natural Language Processing	NLP / text	no	Progressive learning	no	none	NLP
[111]	2020	Progressive Transfer Learning and Adversarial Domain Adaptation for Cross-Domain Skin Disease Classification	Image	no	Progressive learning	yes	none	Medical image
[112]	2017	Progressive Neural Networks for Transfer Learning in Emotion Recognition	Image & audio	yes	Progressive learning	no	none	Para-linguistic

Continuation of Table 3.1								
Ref.	**Year**	**Title**	**Data Type**	**Time Series**	**Approach**	**CNN**	**Known Models Used**	**Dataset Field**
[113]	2020	A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 from chest CT radiography digital images	Image	no	Adversarial-based	yes	Alex-Net, VGG-Net16, VGG-Net19, Google-Net, Res-Net50	Medical image
[114]	2019	Diagnosing Rotating Machines with Weakly Supervised Data Using Deep Transfer Learning	Tabular/ bigdata	yes	Adversarial-based	yes	none	Mechanic
[115]	2017	A New Deep Transfer Learning Based on Sparse Auto-Encoder for Fault Diagnosis	Tabular/ bigdata	yes	Sparse Auto-Encoder	no	none	Mechanic
End of Table								

The second popular approach in DTL is freezing CNN layers in a pre-trained model and finetune only lateral fully connected layers (Freezing CNN layers). CNN layers extract features from the given dataset, and the fully connected layers are responsible for classification, which in this method will be finetuned to the new task for target data.

[99]–[108] are the sample research publications, which have used this method for different data types such as image and tabular data as listed in Table 1. This technique is specific to the models consisting of CNN layers; however, it can be extended to other deep learning models by assuming the earlier and middle layers are acting similar to CNN layers for feature extraction.

Using well-known models such as VGG-Net, Alex-Net, and Res-Net, which has already been trained on ImageNet datasets [116], is a general approach for both of the techniques mentioned

above since they are easily accessible, and they are pre-trained to the highest possible accuracy. It is worth mentioning that such training can take days of processing time even with clusters of GPUs/TPUs and the mentioned methods are skipping the pre-training step by simply downloading a publicly available pre-trained model.

[109]–[112] are based on the progressive learning method, also known as progressive neural networks (PNNs), described earlier. [110] evaluates progressive learning effectiveness for common natural language processing (NLP) tasks: sequence labeling and text classification. Through evaluation and comparison of applying PNNs to various models, datasets, and tasks, they show how PNNs improve DL models' accuracy by avoiding catastrophic forgetting in finetuning techniques. [109], [111], [112] use PNNs for image and audio datasets and similarly finds tangible improvements in comparison to other DTL techniques.

[113] and [114] are examples of adversarial-based approaches that we found in the literature. In [113], they used conditional generative adversarial networks (CGAN) to expand limited target data of chest X-Ray images for detecting Covid-19 DTL model. [114] applies domain adversarial training to obtain the shared features between multiple source datasets.

Moreover, we found some tailored DTL methods for specific tasks and datasets like [115]. The proposed method in [115] as they describe is based on "three-layer sparse auto-encoder to extract the features of raw data, and applies the maximum mean discrepancy term to minimizing the discrepancy penalty between the features from training data and target data." They tailor that method for smart industry fault diagnosis problems and achieve 99.82% accuracy which is better than other approaches like deep belief network, sparse filter, deep learning, and support vector machine. Such tailored DTL approaches are not usually easy to generalize for different tasks or

datasets. Nonetheless, they can open the door to interesting and new techniques in deep transfer learning's future.

3.7 Experimental Analyzations of Deep Transfer Learning

In this section we review two remarkable experimental evaluations of DTL techniques. The tests' setup, analysis, and conclusions are noteworthy for applying DTL techniques in different scenarios.

"What is being transferred in transfer learning?" [117] is a recent experimental study which uses a series of tests on visual domain and deep learning models and tries to investigate what makes a successful transfer and which part of the network is responsible for that. To do so, they analyze networks in four different cases: (i) pre-trained network, (ii) random initialized network, (iii) finetuned network on target domain after pretraining on source domain, (iv) trained network from random initialization [117]. Moreover, to characterize the role of feature reuse, they use a source (pre-train) domain containing natural images (IMAGENET), and a few target (downstream) domains with decreasing visual similarities from natural images: DOMAINNET real, DOMAINNET clipart, CHEXPERT (medical chest X-Rays) and DOMAINNET quickdraw [117].

The study shows that feature reuse plays a key role in deep transfer learning as a pre-trained model on IMAGENET shows the largest performance improvement on real domain, which shares similar visual features (natural images) with IMAGENET in comparison to randomly initialized models. Also, they run a series of experiments by shuffling the image blocks (different block sizes). These experiments prove that feature reuse plays a very important role in transfer learning, particularly when the target domain shares visual features with the source domain. However,

they realize that feature reuse is not the only reason for deep transfer learning success since even for distant targets such as CHEXPERT and quickdraw, they still observe performance boosts from deep transfer learning. Additionally, in all cases pre-trained models converge way faster than random initialized models. [117]

Further, they manually analyze common and uncommon mistakes in the training of randomly initialized versus pre-trained models. They observe that data samples marked incorrect in the pre-trained model and correct in the randomly initialized model are mostly ambiguous samples. On the other hand, the majority of the samples that a pre-trained model marked correct and a randomly initialized model marked incorrect are straightforward samples. This means that a pre-trained model has a stronger prior, and it is harder to adapt to the target domain. Moreover, using centered kernel alignment to measure feature similarities, they conclude that the initialization point drastically impacts feature similarity, and two networks with high accuracy can have a different feature space. Also, they discover similar results for distance in parameter space, which two random-initialized models are farther from each other compared to two pre-trained models. [117]

In regard to performance barriers and basins in the loss landscape, they have concluded that the network stays in the same basin of solution when finetuning a pre-trained network. They reach to this conclusion by training pre-trained models from two random runs as well as training random initialized models twice and comparing. Even when training a random initialized model two times with the same random values the models end up in different basins. [117]

Module criticality is an interesting analysis of deep learning models. Usually, in a deep CNN model each layer of CNN considers a module, while in some models a component of network can be considered as a module. To measure criticality of a module, it is possible to take a trained

model and re-initialize each module at once and compare the amount of model accuracy drop. Adopting this technique, the authors of [117] discovered: (i) fully connected layers (near to model output) become critical for P-T model, and (ii) module criticality increases moving from the input side of model towards output, which is consistent with the concept of earlier layers (near input) extracting more general features while lateral layers have features that are more specialized for the target domain.

[118] is another experimental analysis of transfer learning in visual tasks with the title of "Factors of Influence for Transfer Learning across Diverse Appearance Domains and Task Types". Three factors of influence are investigated in this study: (i) image domain, the difference in image domain between source and target tasks, (ii) task type, the difference in task type, and (iii) dataset size, the size of the source and target training sets. They perform over 1200 transfer learning experiments on 20 datasets spanning seven diverse image domains (consumer, driving, aerial, underwater, indoor, synthetic, closeups) and four task types (semantic segmentation, object detection, depth estimation, keypoint detection). [118]

They use data normalization (e.g., Illumination normalization) and augmentation techniques to improve models' accuracy. They adopt recent high-resolution backbone HRNetV2, which consists of 69M parameters. This backbone is easily adjustable for different datasets by simply replacing the head of the backbone. To make a fair comparison they pre-trained (to be used for transfer learning) their models from scratch and evaluated their performance using top-1 accuracy on the ILSVRC'12 validation set. [118]

The transfer learning experiments are mainly divided into two settings of (i) transfer learning with small target training set and (ii) with the full target set. The evaluation of transfer learning models is based on the gain obtained from finetuning from a specific source model compared

to finetuning from ILSVRC'12 image classification with the main question of "are additional gains possible, by picking a good source?". Furthermore, they added a series of experiments for multi-source training to investigate the impact of using multi-source training for a specific task. [118]

Such an exhaustive experimental analysis resulted in following observations: (i) all experiments proved that transfer learning outperforms training from scratch (random initialization); (ii) for 85% of target tasks there exists a source task which tops ILSVCR'12 pre-training; (iii) the most transfer gain happens when the source and target tasks are in the same image domain (within-domain), which is even more important than source size; (iv) positive transfer gain is possible when the source image domain includes the target domain; (v) although multisource models bring good transfer, they are outperformed by the largest within-domain source; (vi) "for 65% of the targets within the same image domain as the source, cross-task-type transfer results in positive transfer gains"; (vii) as naturally expected, the larger datasets positively transfer towards the smaller datasets; (viii) transfer effects are stronger for a small target training set, which helps the process of choosing the transfer learning model by testing several models with a small section of target data. [118]

3.8 Discussion

The Deep Transfer Learning (DTL) research field is thriving because of the motivation to handle the limitations of Deep Learning (DL) models, which are the dependency on extensive labeled data and training costs. The main idea is to use obtained knowledge from source data in the training process on target data. Another possible impactful outcome of the DTL research line is to achieve continual learning, which brings Artificial General Intelligence [2] a step closer to

reality. Continual learning can be achieved simply through a chain of transfer learning processes while the end model is still valid on all previous training sources.

As we reviewed in previous sections, model-based approaches are the most commonly used approaches in DTL since deep learning models have the capacity to be adjusted to transfer knowledge. However, there are two main constraints in such approaches— catastrophic forgetting dilemma and an overly biased pre-trained model.

In the case of finetuning a pre-trained model, there is a high chance of drastic changes of weights through the whole model resulting in the catastrophic forgetting dilemma. Therefore, the obtained knowledge could be partially or even completely wiped out, resulting in unsuccessful training and no possibility of continual learning. This constraint limits the success of the finetuning approach to tightly related source and target data. Also, a very well-known technique to reduce the forgetting effect is to add a limited number of source samples to the target training data.

Freezing the pre-trained CNN layers technique tries to tackle the catastrophic forgetting by freezing the obtained knowledge on earlier layers and finetuning the fully-connected lateral layers to achieve transfer learning for target data. Given the fact that earlier layers in DL models extract detailed features and move towards the output, more abstract knowledge is extracted [14]; freezing the earlier layers limits the ability of the model to learn any new features from target data, which is known as an overly biased pre-trained model. Having extensive source data or access to a pre-trained model on a large dataset is critical for a successful transfer using this technique. In this way, there is a high chance that the pre-trained model has already learned any possible detailed features, and simply by finetuning the lateral layers can perform on target data. However, even tackling the first obstacle, this solution is still imperiled by the catastrophic forgetting in lateral

layers. This technique is still successful in the case of related source and target data and tasks despite the limitations mentioned above.

Progressive learning tries to find a middle ground between catastrophic forgetting and a biased model by adding a new layer(s) to the end of a frozen pre-trained model. This technique is successful in the case of task transfer for related source and target data. It can not deal with distant source and target data since the earlier layers are frozen and cannot learn new features; however, the new lateral layer helps the model adjust to a new task.

A possible solution to address both catastrophic forgetting and an overly biased pre-trained model in DTL is to increase the learning capacity of a pre-trained model by vertically expanding it. In EXPANSE we propose expanding the model vertically in training on target data, adding new nodes on frozen pre-trained layers throughout the model instead of adding a new layer(s) to the end of the model. The vertical expansion increases the model learning capacity while keeping the previously obtained knowledge intact. Therefore, not only do we achieve successful transfer learning, our final model is still valid on source data opening the door to deep continual learning. [15]

3.9 Conclusion

In this chapter, we review the taxonomy of deep transfer learning (DTL) and the definitions of different approaches. Also, we review, list, categorize and analyze over thirty recent applied DTL research studies. Then, we investigate the methodology and limitations of the three most common model-based deep transfer learning methods: (i) Finetuning, (ii) Freezing CNN Layers, and (iii) Progressive Learning. These techniques have proven their ability and effectiveness for various machine learning problems. The simplicity of finetuning publicly available pre-trained models

on extensive datasets is the reason for it being the most common transfer learning technique. Moreover, two thorough experimental studies in DTL are summarized; their discoveries clarify the details of a successful deep transfer learning approach for different scenarios. Last but not least, the limitations of current DTLs, catastrophic forgetting dilemma, and overly biased pre-trained models are discussed, along with possible solutions.

Chapter 4

EXPANSE (Research Contributions)

4.1 Problem and Motivation

Every advancements starts with a drive/motivation and often has some limitations, which such limitations drive the next advancements. Through the review of AI, ML, and DL studies, we identified such drives and limitations:

- Rule-based AI (early AI) is limited to solved problems, and all possibilities should be hard-wired (program) into an AI agent, e.g., if this then that (IFTTT) applications.
- To overcome the limitations of rule-based artificial intelligence, machine learning models are developed based on statistical and mathematical data analysis to identify patterns for prediction and/or classification. However the traditional ML models are effective on tabular data and linear problems, they cannot handle non-linear data (such as the XOR problem).

- Artificial Neural Networks (ANN) and later Deep Learning (DL) models have been implemented to tackle the limitation of traditional MLs, non-linear problems. Also, later with the development of the Convolutional Neural Network (CNN), such models thrived in solving many more problems such as image-related tasks. However, even DL models come with two significant limitations. First is the dependency on extensive training datasets, usually labeled data, which is almost impossible to gather in some areas such as medical images. Second, the training cost, the training process of a DL model needs a tremendous amount of processing power, and it takes a long time.

- Transfer Learning in Deep Learning, known as Deep Transfer Learning (DTL), started to bloom with the goal of addressing DL models limitations. The idea behind DTL is to reuse obtained knowledge in a model from a source data/task for a target data/task. We have done a systematic literature review (chapter 3) to understand the current advancements in DTL and their possible limitations.

Even though there are many successful studies in the deep transfer learning field, the methods are mostly tailored to a specific dataset and a task, and the success is mainly achieved based on trial and error. As mentioned in the introduction section and chapter 3, most existing deep transfer learning (DTL) methods are network/model-based approaches and can be categorized into three groups [4].

The first category is based on finetuning a pre-trained model. In this case, a related pre-trained model, trained on source data, will be used and trained (finetuned) on the target data. The primary issue in this approach is known as catastrophic forgetting dilemma since the obtained knowledge can be partially or totally wiped out [5], [6], making it a foremost necessity for this method to use similar source and target data. Catastrophic forgetting happens because the weights throughout

the model can drastically change in the process of finetuning on target data [4], [6]. However, if the source and target data are close enough, there is a high chance of success [119]–[122].

The second category is mainly for deep CNN models. In this method, the pre-trained CNN layers are frozen for the target domain training process with a new header and lateral fully-connected layers [4]. Change of header helps adjust the input shape, and change to the fully-connected layers brings the possibility of changing the model objectives, while the frozen CNN layers do the feature extractions from details to abstract. This technique reduces the effect of catastrophic forgetting to some extent. However, the model is still strongly biased on source data and cannot adapt to target data. Success depends on whether the source and target data are tightly related. Therefore, to increase the success rate, it is a known practice to choose a profoundly robust pre-trained model on a massive dataset such as ImageNet [123] to ensure that source data is broad enough to contain target data to some extent. Such pre-trained models are available publicly [123], which is why this method has been used extensively. [124]–[128] are successful examples of using this method.

The third category of model-based approaches is progressive learning, which aims to address both catastrophic forgetting and adaptability to target data and task. In progressive learning, the whole pre-trained model is used frozen while a new layer(s) adds to the end of the model for the process of training on target data [8]. This approach can deal better with slightly less related source and target data. However, it still suffers from being biased on source data when the target has more features or different features than the source. Some successful examples of using this approach are [9]–[12].

The closest study to EXPANSE that we found is [129], which is an intriguing study of expanding the CNN model both horizontally and vertically in the process of finetuning. They try to evaluate and compare the effects of expanding the model in two ways, adding new layers vs. adding

units on existing layers. Their results prove that expanding the model (or "growing" the model, as they call it) in either direction is helpful in the transfer of knowledge and obtaining better accuracy on target data. They conclude that adding units on some layers instead of adding new layers has a slight but consistent benefit in their experiments, which aligns with our EXPANSE assumptions even though their study's mindset and evaluation setup are not the same. However, they only explore finetuning without freezing any part of the pre-trained model, even theoretically. Also, their method is specific to CNN layer expansion and only on some lateral layers, which, as we mentioned, results in a biased pre-trained model without the capacity to learn any new detailed features (on earlier layers).

Furthermore, while studying DL and DTL techniques, we have realized that no research address or pays attention to the different samples' quality in training datasets. In every study, training DL models are based on random initialization of weights in the model and using a mixed distribution of training data. The finding of this gap initiated the concept of the two-step training method in EXPANSE.

To our knowledge, all training processes in machine learning and deep learning are based on a mixed distribution (mostly normal distribution) of the training dataset. We did not find any approach that divides the training data based on the quality of samples, divides the training process into steps, or adjusts the learning rate accordingly. All deep learning models are initialized by random values and start extracting features from details to abstract (first layers towards output layer) [14] with no prior knowledge. An exaggerated translation of this approach would be like teaching a child to identify cancer tissues by showing thousands of images with cancer tissues marked.

In the process of training the DL model, the learning rate adjusts the size of the steps in the gradient descent on the error surface [130], [131]. It is a tricky parameter (hyper-parameter) since if it is too small, the model may never move enough on the error surface to find the local (global) minima. On the other hand, the model may jump over local minima basins and never find them if the step is too big. Numerous studies have been done on this topic, e.g., variable learning rate [130], [132]. However, we could not find any study that adjusts the learning rate based on the quality of training data in deep transfer learning.

4.2 EXPANSE

EXPANSE's design is based on the human education system and is inspired by progressive learning [8]. The main objective is to improve deep transfer learning (DTL) by dividing the problem and the associated model into more straightforward and simple steps. Then, through a continual/progressive learning approach, gradually expand the model and training samples. Thus, we named our proposed system EXPANSE. [13]

The first aspect of EXPANSE is about dividing the training samples. We will add a limited number of perfected samples or select limited exemplary samples from the training dataset depending on the training data. We train the model first with the exemplary samples at a higher learning rate (LR), then finetune that model with the whole training data (containing exemplary samples as well) at a lower learning rate (LR). We call this approach two-step training. This technique aims to help the model extract more meaningful features and find a finer basin on the error surface with larger steps (higher learning rate) using a limited number of exemplary samples; then explore that area of the error surface to find the local minima with smaller steps (lower learning rate).

The two-step training method follows our learning process in schools. We first learn the fundamentals, then navigate uncertainties and more complex problems. We have been trained and practiced basic knowledge for years from the beginning of K-12 classes. After a long time of learning and reaching a solid basic knowledge, more complex problems add to our learning agenda near the end of the K-12 system. Still, even in the early years of college and bachelor's degrees, we mostly learn about basics, and the problems extend with a limited amount of uncertainties and complexities. Mostly, more complex problems are limited to graduate degrees. This long-life process of learning basics helps us not deviate from fundamentals in the exploration of new ideas. We believe that in DL models, it is possible to direct the model to find a more meaningful basin on the error surface with the hope that the model will not deviate from it in adding more complex samples or during the finetuning process.

The other aspect of EXPANSE is about model expansion in the process of deep transfer learning. Increasing the model's size in either direction increases the model's learning capacity. As we mentioned, the earlier layers in deep learning extract detailed features. And moving towards the output, the layers are responsible for extracting abstracts by the obtained detailed features [14]. Therefore, if the target domain contains more features than the source domain, adding layers towards the end of the model is not enough. The model cannot extract those detailed features on a frozen model or replace the previously gained knowledge on earlier layers. As we mentioned in the previous section, these two extremes are known as an overly biased pre-trained model with frozen layers or catastrophic forgetting dilemma in the case of finetuning.

To address this, in EXPANSE, we consider that the model can expand in both directions (new nodes on existing layers and new layers) while the pre-trained section of the model will be used frozen. The expansion of layers (adding new nodes) will increase the learning capacity of the

network. The earlier layers can extract more detailed features, and lateral layers can extract more complex abstract knowledge, as well as the output layer. Adding the new lateral layer(s) becomes crucial when the model objective changes drastically towards more complex tasks in the process of deep transfer learning. Figure 4.1 illustrates the vertical model expansion in EXPANSE. The continual learning in EXPANSE happens first because the expanded model stays valid for source data. The final model can also be again expanded for another step of training on new target data, and this process can be repeated. The algorithm 1 shows the EXPANSE methodology's algorithm.

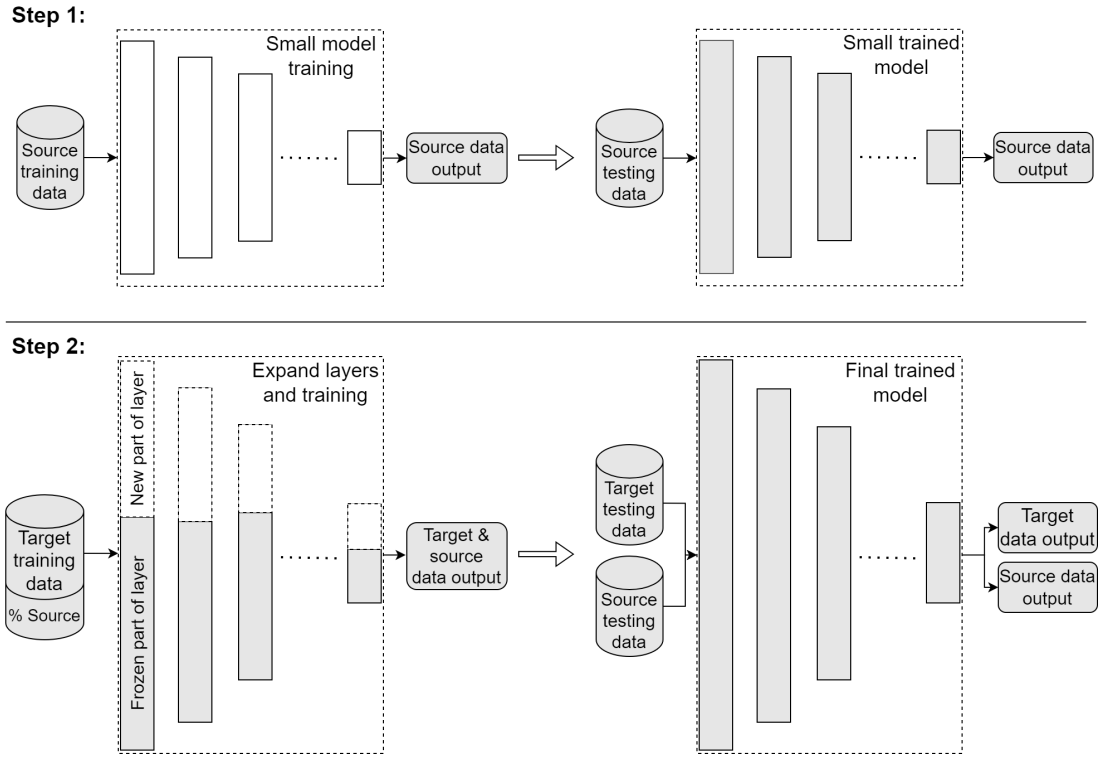

Figure 4.1: Model expansion illustration.

Decision-making regarding model expansion size is similar to the deep learning model hyper-parameter adjustments. There is no exact formula, but through experience and analysis of data

Algorithm 1 EXPANSE

Require: source data, target data, exemplary source data, exemplary target data
$ModelA \leftarrow Firstmodel$ for source data (random initialized)
$ModelB \leftarrow Train$ A on exemplary source data
$ModelC \leftarrow Finetune$ B on exemplary source data + source data
$ModelD \leftarrow Expand$ C with new nodes (random initialized) on layers
$ModelE \leftarrow Freeze$ the C part of model D
$ModelF \leftarrow Train$ E on exemplary target data
$ModelFinal \leftarrow Finetune$ F on exemplary target data + target data
$return(ModelFinal)$

and tasks, an expert can narrow down the size of the model expansion and the parameters in a limited number of trials. In EXPANSE, the expansion size should be decided by common sense and experience based on the difference in tasks and training data in each step of deep transfer learning.

A key to successful expansion is to consider that the earlier layers need expansion when the target data offers more detailed features, and the middle layers should expand when we expect the model abstract knowledge extraction to increase. Lateral layers can be expanded when we expect to extend the decision-making options in the model, e.g., more classes to identify. Also, there is the option to add new layers to the end of the model in case of task/objective transfer, similar to progressive learning [8].

Continual learning is a game changer in AI since it opens the possibility of learning in continuous form for an AI. As humans, we learn skills and knowledge on top of previously learned ones. We usually do not forget our previous knowledge and skills; even if we do, it is easily possible to remember them with some reviews. Continual learning in deep learning can be achieved when the final model trained on target data is still valid on source data. We believe in EXPANSE tack-

ling the catastrophic forgetting dilemma by increasing the model learning capacity makes deep continual learning accessible.

The EXPANSE is a methodology to be applied to different problems and DL models, not a solution for a specific case. We are introducing a systematic continual learning approach to deep transfer learning. The goal is to improve the process of transfer learning in deep learning and open a door to a more sophisticated approach to continual learning to move the current AIs a step closer to AGI (human-level continual/progressive learning).

Chapter 5

Evaluation

EXPANSE is a methodology applicable to machine learning models such as deep neural networks and deep CNN models. To evaluate such a methodology, we need to verify its practicality and performance with more general tasks and compare results to a similar model and task without the EXPANSE methodology applied. Such evaluation aims not to prove that this approach can outperform other techniques for a singular task; as long as the method is applicable and results are similar to a model without EXPANSE, we have achieved our goal. The goal is to open the door to continual learning possibility, not compete on a single task. However, our results proved the applicability and practicality of EXPANSE and showed unanticipated performance improvements.

One major limitation of the proposed system is the implementation of adding new nodes to frozen layers since it requires a structural change to the model's core. We need to be able to freeze part of a layer and let the new nodes be trainable on the target data on that layer. Today, none of the existing framework's libraries for DLs supports such an ability. Also, in the case of the CNN layer, this implementation is even more complicated; however, it might be dealt with by increasing the number of channels in that layer and only freezing the pre-trained channels in that layer.

To evaluate EXPANSE, we use deep neural networks, and instead of freezing the pre-trained section, we finetune the expanded model. The Algorithm 2 shows the EXPANSE adjusted algorithm, including this limitation. We consider dealing with this technical implementation issue of freezing part of a layer as future work. However, the results show the potential of this methodology even without freezing and demonstrate that being able to freeze the pre-trained section should increase the performance of EXPANSE in the future.

Algorithm 2 EXPANSE (adjusted)

Require: source data, target data, exemplary source data, exemplary target data
$ModelA \leftarrow Firstmodel$ for source data (random initialized)
$ModelB \leftarrow Train$ A on exemplary source data
$ModelC \leftarrow Finetune$ B on exemplary source data + source data
$ModelD \leftarrow Expand$ C with new nodes (random initialized) on layers
$ModelE \leftarrow Freeze$ the C part of model D
if E is done **then** (future work)
    $ModelF \leftarrow Train$ E on exemplary target data
    $ModelFinal \leftarrow Finetune$ F on exemplary target data + target data
else if E is NOT done **then** (current evaluation)
    $ModelFinal \leftarrow Finetune$ D on target data + exemplary target data + exemplary source data + a portion of source data
end if
$return(ModelFinal)$

We consider size and configuration of the end model (after expansion) as the final model to have a benchmark for comparison. We train the final model from randomly initialized weights and consider the obtained results as the benchmark. We save our random initialized weight and start all the training with the same weights to make a fair comparison. Also, through all tests, we keep the hyper-parameters constant. All experiments are easily reproducible since we have used random seeding so that any run will end up with the same results. The experiments were done on Google Colab using python and TensorFlow (Keras) libraries [133], accessible at [134].

We chose the MNIST dataset [135], [136] and a deep neural network model similar to [136], [137] for our evaluation process. This dataset consists of 60,000 handwritten digits for the training set and 10,000 samples for testing, see figure 5.1. MNIST is a well-known dataset in the Machine Learning (ML) community to benchmark new methods [138]. Also, it is possible to create exemplary perfected data using printed digits for evaluating our proposed two-step training.

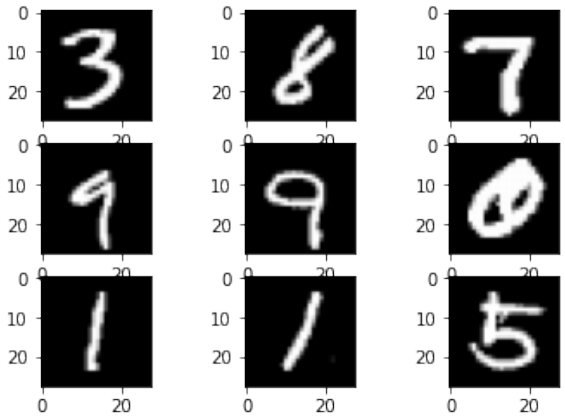

Figure 5.1: MNIST dataset samples.

For this experiment, we made 180 perfected samples of ten printed digits using 18 fonts, and by simply duplicating the samples, we made a perfected dataset of 360 samples, see figure 5.2. Our final model consists of three (3) layers of 256, 128, and 10 nodes, using Relu activation function, Adam optimizer, and 10-fold cross-validation [2], [139]. Our experiments' epoch number is limited since we use 10-fold cross-validation; one epoch means nine (9) times running through the training data.

In the following experiments, we first evaluate two-step training separately and then evaluate the EXPANSE methodology: model expansion and two step-training.

To verify the effectiveness of the two-step training method, we train the final model with the random initialized value with two-step training; first, train on perfect data and then add

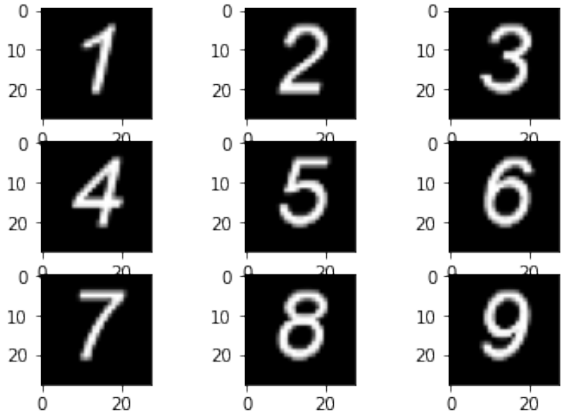

Figure 5.2: Perfected samples.

handwritten samples to compare the result with the benchmark. Moreover, we trained the final model with the randomly initialized weights by mixing the perfect data and the MNIST training dataset.

Table 5.1 shows the obtained results from the two-step training evaluation. As listed in the first two rows (the benchmarks), after getting trained by 60,000 handwritten samples, the model can only be 57.22% accurate in detecting printed digits in the best case. The expectation from such training is that the model should easily identify perfect samples, but the result aligns with our assumption that the usual training process of deep learning is not always meaningful or capable of building on fundamental knowledge. On the other hand, the model trained by only 360 (0.6% of the size of MNIST training dataset) exemplary perfect data samples in less than 30 seconds can detect around 40% of handwritten samples (row number 3). Row number 4 shows the results of finetuning the model from row number three on only the training dataset of MNIST, which shows the catastrophic forgetting dilemma since the accuracy of perfect data dropped. To avoid the

catastrophic forgetting dilemma, we add the perfect samples to the target data (MNIST training data) for the finetuning process (row number 6).

The accuracy of the two-step trained model implies that there existed a finer local minima on the error surface, which could work for both the perfected data and the handwritten data to improve the accuracy of handwritten test data. In other words, the model found a more meaningful basin on the error surface. Also, even adding the perfect samples to the MNIST training data for training the randomly initialized model in one step (row number 5) while improving the results is still not as accurate as two-step training (row number 6). This follows the logic behind the human education system: the most effective way of learning is to focus first on the basics and then add complexity.

It is worth mentioning that increasing the number of epochs in these tests empowered the effectiveness of the two-step method. We limited the number of epochs to limit the training time for the purpose of reproducibility and the verification process. For instance, the same experiment with eight (8) epochs resulted in 98.33% (two-steps training) vs. 98.07% (traditional training) accuracy on the MNIST test dataset.

To verify the applicability and effectiveness of model expansion along with two-step training, the EXPANSE system, we divided the MNIST training data and exemplary data into two sets of 0 to 4 digits and 5 to 9 digits. The MNIST training data consists of 30,596 samples of 0 to 4 and 29,404 samples of 5 to 9 digits. We reduced the final model size to three (3) layers of 150, 80, and 5 nodes for the first training step. First, we trained the small model using the 180 exemplary samples and then finetuned it with a mix of MNIST training data and exemplary samples for digits 0 to 4. For applying the model expansion, we updated the weights of the small portion of the random initialized final model (three (3) layers of 256, 128, and 10 nodes) with the weights of a finetuned

Table 5.1: Two-step training on MNIST

#	Description	Accuracy on		
		Exemplary data	MNIST train data	MNIST test data
1	Random initialized on MNIST with LR=0.001 & epoch=3	55.56%	99.93%	98.04%
2	Random initialized on MNIST with LR=0.002 & epoch=3	57.22%	99.57%	97.83%
3	Random initialized on Perfect data with LR=0.01 & epoch=8	100.00%	39.54%	41.24%
4	Fine-tune the pre-trained model on only MNIST with LR=0.002 & epoch=3	68.33%	99.93%	97.97%
5	Random initialized on Mix data with LR=0.002 & epoch=3	100.00%	99.71%	98.02%
6	(Two-steps training) Fine-tune the pre-trained model on Mix data with LR=0.002 & epoch=3	100.00%	99.79%	98.06%

small model. Then we trained that model using the mix of MNIST training data and exemplary samples of 5 to 9 with a part of the samples from 0 to 4. We mixed some samples of the previous step into the final step since we could not freeze the small section of the network, and we wanted to avoid catastrophic forgetting.

Table 5.2 shows the obtained results at each step of this process. The first two rows are only for samples of 0 to 4 digits. As listed on row number 4, EXPANSE methodology has improved the same model's (size and configuration) accuracy from 98.04% (Table 5.1, row number 1) to 98.09% on MNIST test data and from 55.56% to 100% on printed digits. This demonstrates the practicality and effectiveness of the EXPANSE continual learning system and shows that this methodology can improve the performance of the same model (size and configuration).

Table 5.2: EXPANSE on MNIST

#	Description	Accuracy on Exemplary data	Accuracy on MNIST train data	Accuracy on MNIST test data
1	Small model, random initialized on perfect (0 to 4) with LR=0.01 & epoch=3	100.00%	64.62%	NA
2	Small model, fine-tune on mix data (0 to 4) with LR=0.002 & epoch=3	100.00%	99.97%	NA
3	Final model (loaded weights from small model), fine-tune on mix data with LR=0.001 & epoch=3	100.00%	99.89%	98.04%
4	Final model (loaded weights from small model), fine-tune on mix data with LR=0.002 & epoch=3	100.00%	99.87%	98.09%

A key point in this set of experiments is that the digits 0, 1, 2, 3, and 4 do not share the same visual features as 5, 6, 7, 8, and 9. Therefore, in this case, the source and target data are not closely related and can be considered distant datasets, and EXPANSE successfully handled it.

Further, we selected the best deep neural network (DNN) model without any pre-processing (e.g., distortion techniques) of training data from the list at [135] to verify if, by applying EXPANSE, we can improve such a model. The selected model is listed as "3-layer NN, 500+300 HU, softmax, cross entropy, weight decay", with an accuracy of 98.47%. We first implemented their model and obtained 98.46% accuracy on MNIST test data, table 5.3 row number 2. Then, we applied EXPANSE to the same model, the first step with a smaller model of 300, 200, and 5 nodes and the second step of 500, 300, and 10 nodes. We obtained 98.52% accuracy, table 5.3 row number 3. Not only do we improve the accuracy of the existing model on MNIST test data, but our model can also detect printed digits perfectly. It is worth mentioning that a pre-processing technique or use of CNN layers is necessary to gain accuracy above 99% on MNIST [140].

Table 5.3: EXPANSE on MNIST vs. best existing deep neural network (DNN) model

#	Description	Accuracy on		
		Exemplary data	MNIST train data	MNIST test data
1	Random initialized on MNIST with LR=0.0009 & epoch=5	58.33%	99.99%	98.42%
2	Random initialized on MNIST with LR=0.0009 & epoch=7	53.89%	99.99%	98.46%
3	Final model (loaded weights from small model), fine-tune on mix data (66% of source data) with LR=0.0009 & epoch=5	100.00%	99.75%	98.52%
4	Final model (loaded weights from small model), fine-tune on mix data (10% of source data) with LR=0.0009 & epoch=5	100.00%	97.65%	96.55%

In this experiment, we used two-thirds of the source data mixed with target data to avoid catastrophic forgetting for the second step. However, even with using only 10% of source data mixed with target data for the second step of training, we achieved 96.55% accuracy on the MNIST test dataset, table 5.3 row number 4. This very slight drop in accuracy shows that even without being able to freeze the small part of the trained model after the first step, the model is not dramatically losing the obtained knowledge from source data while getting trained on target data. In our opinion, the reduction of the catastrophic forgetting effect in EXPANSE is because vertically expanding the model provides a capacity to learn new features without replacing the previously obtained knowledge. Moreover, this implies that by being able to freeze the smaller section of the model in the second step (future work), we will reduce training time and possibly improve accuracy since we will not need to use any source data (or a limited number of samples) during the training on target data.

Following the promising results on MNIST, we also applied EXPANSE to a simple model of a three-layer (500, 300, and 10 nodes) neural network on Fashion MNIST [141]. Fashion MNIST dataset "comprising of 28x28 grayscale images of 70,000 fashion products from 10 categories, with 7,000 images per category. The training set has 60,000 images and the test set has 10,000 images, see figure 5.3. Fashion-MNIST is intended to serve as a direct drop-in replacement for the original MNIST dataset for benchmarking machine learning algorithms, as it shares the same image size, data format and the structure of training and testing splits." [141]

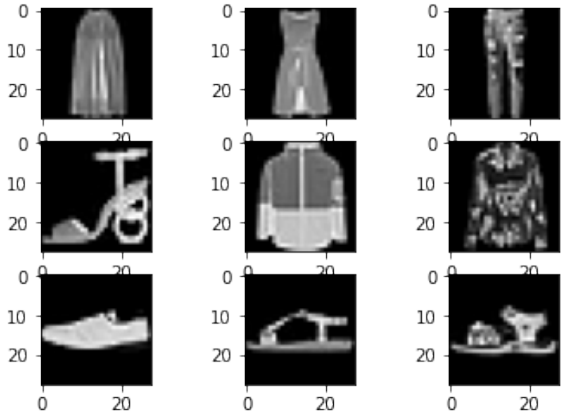

Figure 5.3: Fashion MNIST dataset samples

In this case, instead of making perfected samples, we manually selected a limited number of exemplary samples from the training dataset (18 samples for each of the 10 categories) based on the simplicity and clarity of the samples, see figure 5.4. Like previous experiments, we started with a smaller model (300, 200, and 5 nodes) and the first five categories. Next, we expanded the model to 500, 300, and 10 nodes and trained on all categories. Again, we use a part of the first step's training samples in the second step to reduce the catastrophic forgetting effect. Also, similar to previous tests, we trained the randomly initialized final model on the F-MNIST training dataset as the benchmark to compare our results with it.

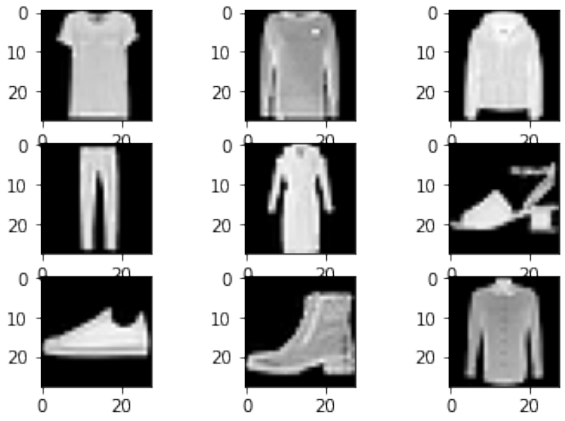

Figure 5.4: Fashion MNIST exemplary selected samples

The results on F-MNIST are consistent with our earlier MNIST experiments; both two-step training and model expansion achieved better accuracy as well, table 5.4 row 1 and 2 vs. 3 and 4. Even using a set of exemplary data from the same training set and applying two-step training improved the model accuracy. Moreover, similar to MNIST experiments reducing the reuse of source data in the target training process to 10% did not drastically drop the accuracy, table 5.4 row number 5.

These results convey the capability of EXPANSE in continual learning, meaning that EXPANSE can be applied to a chain of problems, and the final model can answer all the problems and reuse the knowledge obtained at every stage of training. Moreover, the models from the intermediate steps of such a chain of training could be used later for different tasks. For example, a final task of image classification of interior objects could be done through steps of 1) simple shapes, 2) colors, 3) furniture, 4) building objects, 5) arts, 6) plants, etc. So, the model from step 4, building

Table 5.4: EXPANSE on Fashion MNIST

#	Description	Accuracy on		
		Exemplary data	F-MNIST train data	F-MNIST test data
1	Random initialized on F-MNIST with LR=0.0001 & epoch=6	96.67%	96.79%	88.82%
2	Random initialized on F-MNIST with LR=0.0001 & epoch=7	98.89%	97.86%	89.24%
3	Final model (loaded weights from small model), fine-tune on mix data (70% of source data) with LR=0.0001 & epoch=6	100.00%	98.43%	89.42%
4	Final model (loaded weights from small model), fine-tune on mix data (70% of source data) with LR=0.0001 & epoch=7	100.00%	98.73%	89.43%
5	Final model (loaded weights from small model), fine-tune on mix data (10% of source data) with LR=0.0001 & epoch=6	100.00%	90.24%	84.39%

objects (door, window, stairs, etc.) could be separately used (trained on more data) for interior and exterior building object classification.

All the implementations are available and reproducible at [134].

5.1 Discussion

EXPANSE is a new methodology for deep transfer learning, which opens the possibility of continual deep learning while tackling the catastrophic forgetting dilemma and the overly biased pre-trained model issues in current DTL techniques. The main three contributions of this methodology are: (i) introducing the two-step training, which is an educational approach in the process of

training a model, (ii) increasing model learning capacity in the vertical direction to deal with both catastrophic forgetting dilemma and overly biased pre-trained model in DTLs, and (iii) opening a door to continual deep learning since the final model using this system is still valid on source data.

In the previous section, we described the details of our evaluation steps with detailed results; here, we are going to summarize those results and see how they prove the validity of claimed contributions mentioned above. It is worth reminding that all our evaluations setup have been designed to make the most meaningful and fair comparison since we have used the same random initialization values and hyper-parameters for all the experiments. Also, all the implementations are publicly available at [134], and the results are easily reproducible.

Given the evaluations and summarized results of the two-step training test, table 5.5, the two-step training shows its great potential in improving the training process of deep learning models. The observations suggest that the model finds a more meaningful basin on the error surface by getting trained first with exemplary samples. The additional training step is almost at no cost (process or time) since the number of exemplary samples could be very limited. Another important observation from this evaluation is that the two-step training reduced the overfitting issue of the model since while the accuracy improved on test data, it actually reduced on training data, table 5.5. Moreover, using the limited exemplary data, the model was able to detect real data (test data) with an intriguing level of accuracy. In contrast, the model trained by extensive real data could not perform well in classifying exemplary (basic) samples.

Tables 5.6, 5.7, and 5.8 summarize the evaluations of EXPANSE, two-step training along with model expansion, in three different scenarios. First a model on MNIST dataset, then a competitive test on MNIST, and last a model on Fashion MNIST. EXPANSE applies in three steps: (i) the small model is trained on exemplary source data, (ii) the small pre-trained model is finetuned with

Table 5.5: Two-step training on MNIST, summary

#	Description	Accuracy on		
		Exemplary data	F-MNIST train data	F-MNIST test data
1	Baseline: Random initialized on MNIST with LR=0.001 & epoch=3	55.56%	99.93%	98.04%
2	Two-step training: First trained on exemplary data with LR=0.01, then finetune on mix of data with LR=0.002 & epoch=3	100.00%	99.79%	98.06%

a mix of exemplary and training source data, and (iii) the small pre-trained model is expanded with new nodes to the size of the baseline model (final model) and is trained (finetuned) on a mix of target data and a portion of source data.

An interesting observation is that in all three tests, the model accuracy on the training data has dropped using EXPANSE while improving on test data. This means that the EXPANSE is reducing the overfitting issue during the training. Also, it is clear that using the model expansion improves the accuracy of the final target data while the final model is still valid on source data. This shows the EXPANSE's potential for deep continual learning. To our knowledge, all other DTL techniques are focused on performing on target data, and their final model is useless on source data.

The results in three different models and two different datasets are consistent. EXPANSE over-performed the baseline in all experiments while the final model stayed valid on source data. Also, on the dataset with more complex data, the improvement achieved by EXPANSE is even more tangible. This proves the potential behind the model expansion of EXPANSE. To our

knowledge, this is the first attempt to successfully expand a pre-trained model vertically (adding nodes to layers instead of layers to model) on every layer. Even though we finetune the expanded model on target data without freezing the small model (implementation limitation), the results are still promising and align with our design expectations.

Table 5.6: EXPANSE on MNIST, summary

#	Description	Accuracy on		
		Exemplary data	F-MNIST train data	F-MNIST test data
1	Baseline: Random initialized on MNIST with LR=0.001 & epoch=3	55.56%	99.93%	98.04%
2	EXPANSE: Two-step training and model expansion in three steps with final step's LR=0.002 & epoch=3	100.00%	99.87%	98.09%

Table 5.7: EXPANSE on MNIST vs. best existing DNN model, summary

#	Description	Accuracy on		
		Exemplary data	F-MNIST train data	F-MNIST test data
1	Baseline: Random initialized on MNIST with LR=0.0009 & epoch=7	53.89%	99.99%	98.46%
2	EXPANSE: Two-step training and model expansion in three steps with final step's LR=0.0009 & epoch=5	100.00%	99.75%	98.52%

These experiments' source and target data have tangible differences in their visual features and could be considered distant datasets. As we expected, the vertical expansion of all layers, specifically the earlier layers (near the input), increased the model learning capacity even for extracting new

Table 5.8: EXPANSE on Fashion MNIST, summary

#	Description	Accuracy on		
		Exemplary data	F-MNIST train data	F-MNIST test data
1	Baseline: Random initialized on F-MNIST with LR=0.0001 & epoch=7	98.89%	97.86%	89.24%
2	EXPANSE: Two-step training and model expansion in three steps with final step's LR=0.0001 & epoch=7	100.00%	98.73%	89.43%

detailed features from data in case of distant source and target data since the final model kept the knowledge extracted from source data and added knowledge from target data. Furthermore, expanding lateral layers (near the output) worked well in adding more classification categories to the model.

In summary, EXPANSE tackled the catastrophic forgetting dilemma since the final model is still valid and performs well on source data. Also, EXPANSE handled transfer learning between distant source and target datasets, which means it overcomes the overly biased pre-trained model issue in deep transfer learning. The two-step training reduced the overfitting issue in deep learning training and improved the quality of learning and the final model accuracy. Last but not least, the EXPANSE methodology can be used for deep continual learning since the final model is valid on both source data and target data.

EXPANSE is a fundamental change in deep learning and deep transfer learning. From the evaluation process, we observed accuracy improvements. We validated that the proposed system is applicable, practical, and effective. A key point about EXPANSE is the potential for continual

learning since the final model is still valid on source data. The real impact of the proposed system is when the target data is scarce or the final task complexity is high. The EXPANSE system can be considered a significant step towards artificial general intelligence (AGI) since it follows the methodology of continual/progressive learning and implies that learning perfection first then improves knowledge gain when uncertainties are introduced.

Chapter 6

Conclusion

In this dissertation, we review the history of AI development from rule-based to machine learning, followed by deep learning. In each step of these developments, we reviewed their limitations and the drive for the next step. Briefly, rule-based AI is limited to solved problems, and all possibilities should be hard-wired (programmed) into an AI agent. Machine Learning (ML) models were developed to overcome the limitation of rule-based AIs based on statistical and mathematical data analysis to find patterns with the goal of prediction and/or classification. Traditional ML models are successful on tabular data and linear problems but cannot address non-linear problems (e.g., XOR problem).

Artificial Neural Networks (ANN) and later Deep Learning (DL) models have been implemented to tackle the limitation of traditional MLs: non-linear problems. Moreover, with the development of the Convolutional Neural Network (CNN), such models became much more competent in solving image-related problems. In spite of this, even DL models have two significant limitations. The first problem is the requirement for extensive training datasets, usually labeled data, which are nearly impossible to obtain in certain areas, such as the medical industry.

The second problem is the training costs. DL models require a great deal of processing power, and they take a long time to train.

Deep Transfer Learning (DTL), Transfer Learning in Deep Learning, began to blossom in response to the limitations of Deep Learning models. In DTL, knowledge obtained from a source file/task is reused in a model for a target file/task. We have done a systematic literature review to understand the current advances in DTL and their likely restrictions. In short, we concluded from our thorough review that most current DTL approaches are model-based approaches using finetuning and/or freezing layers in DL models. Even though many successful studies of such methods have been done for numerous problems, they either suffer from the catastrophic forgetting dilemma or an overly biased pre-trained model because of the nature of using fine-tuning or freezing techniques.

Moreover, we have discovered that there is little research addressing or taking into account the quality of different samples in training datasets in the context of DL and DTL techniques. The weights in the trainer's model are randomly initialized, and mixed distributions of training data are used in every study. In EXPANSE, this gap led to the concept of a two-step training method.

Inspired by the human education system, we introduce the two-step training method for deep learning models: training the model first with limited exemplary data (learning basics first), then fine-tuning that model with the training data (learning to deal with uncertainties and complexity). This method can be considered an educational supervised learning. Our evaluation demonstrated the effectiveness of using two-step training compared to the traditional training of deep learning models.

Furthermore, vertical model expansion, along with two-step training, the EXPANSE methodology has been introduced to deal with catastrophic forgetting dilemma and overly biased models

in deep transfer learning by increasing the whole model's learning capacity through vertical model expansion. Adding new nodes to the pre-trained layers increases the model's learning capacity, even when extracting detailed features for unrelated source and target data. Moreover, this approach opens a reliable path to continual learning in deep transfer learning since our final model is still valid on source data. Even with implementation limitations (freezing part of a layer), our evaluation demonstrates the high potential of the EXPANSE approach for a successful transfer in deep learning that can be applied to different DL models and tasks.

Implementing a library of deep learning to handle freezing a portion of nodes in a layer of the DL model is a substantial future work of our study, which can be handled with a team of developers in a timely manner. Similarly, integrating the EXPANSE methodology in models with CNN layers is another intriguing challenge.

Bibliography

[1] S. J. Russell and P. Norvig, *Artificial intelligence: A modern approach. malaysia*, 2016.

[2] M. Iman, H. R. Arabnia, and R. M. Branchinst, "Pathways to artificial general intelligence: A brief overview of developments and ethical issues via artificial intelligence, machine learning, deep learning, and data science," *Advances in Artificial Intelligence and Applied Cognitive Computing*, pp. 73–87, 2021.

[3] A. Farahani, S. Voghoei, K. Rasheed, and H. R. Arabnia, "A brief review of domain adaptation," *Advances in Data Science and Information Engineering*, pp. 877–894, 2021.

[4] M. Iman, K. Rasheed, and H. R. Arabnia, "A review of deep transfer learning and recent advancements," *arXiv preprint arXiv:2201.09679*, 2022.

[5] X. Chen, S. Wang, B. Fu, M. Long, and J. Wang, "Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[6] C. V. Nguyen, A. Achille, M. Lam, T. Hassner, V. Mahadevan, and S. Soatto, "Toward understanding catastrophic forgetting in continual learning," *arXiv preprint arXiv:1908.01091*, 2019.

[7] H. S. Chang, M. C. Fu, J. Hu, and S. I. Marcus, "Google deepmind's alphago: Operations research's unheralded role in the path-breaking achievement.," *OR/MS Today*, vol. 43, no. 5, pp. 24–30, 2016.

[8] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, *et al.*, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[9] A. Moeed, G. Hagerer, S. Dugar, *et al.*, "An evaluation of progressive neural networksfor transfer learning in natural language processing," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 1376–1381.

[10] Y. Gu, Z. Ge, C. P. Bonnington, and J. Zhou, "Progressive transfer learning and adversarial domain adaptation for cross-domain skin disease classification," *IEEE journal of biomedical and health informatics*, vol. 24, no. 5, pp. 1379–1393, 2019.

[11] J. Gideon, S. Khorram, Z. Aldeneh, D. Dimitriadis, and E. M. Provost, "Progressive neural networks for transfer learning in emotion recognition," *arXiv preprint arXiv:1706.03256*, 2017.

[12] R. Quan, Y. Wu, X. Yu, and Y. Yang, "Progressive transfer learning for face anti-spoofing," *IEEE Transactions on Image Processing*, vol. 30, pp. 3946–3955, 2021.

[13] M. Iman, K. Rasheed, and H. R. Arabnia, "Expanse, a continual deep learning system; research proposal," *International Conference on Computational Science and Computational Intelligence*, 2022.

[14] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," *arXiv preprint arXiv:1506.06579*, 2015.

[15] M. Iman, J. A. Miller, K. Rasheed, R. M. Branchinst, and H. R. Arabnia, "Expanse: A deep continual/progressive learning system for deep transfer learning," *arXiv preprint arXiv:2205.10356*, 2022.

[16] A. M. Turing, "Computing machinery and intelligence," in *Parsing the turing test*, Springer, 2009, pp. 23–65.

[17] J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon, "A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955," *AI magazine*, vol. 27, no. 4, pp. 12–12, 2006.

[18] S. Harnad, "Other bodies, other minds: A machine incarnation of an old philosophical problem," *Minds and Machines*, vol. 1, no. 1, pp. 43–54, 1991.

[19] P. Schweizer, "The truly total turing test," *Minds and Machines*, vol. 8, no. 2, pp. 263–272, 1998.

[20] J. Howard, "Artificial intelligence: Implications for the future of work," *American Journal of Industrial Medicine*, vol. 62, no. 11, pp. 917–926, 2019.

[21] F. Hayes-Roth, "Rule-based systems," *Communications of the ACM*, vol. 28, no. 9, pp. 921–932, 1985.

[22] E. J. Horvitz, J. S. Breese, and M. Henrion, "Decision theory in expert systems and artificial intelligence," *International journal of approximate reasoning*, vol. 2, no. 3, pp. 247–302, 1988.

[23] I. Graham and P. L. Jones, *Expert systems: knowledge, uncertainty, and decision*. Chapman & Hall, Ltd., 1988.

[24] M. Moed and G. Saridis, "A boltzmann machine for the organization of intelligent machines," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 5, pp. 1094–1102, 1990. DOI: 10.1109/21.59972.

[25] R. Salakhutdinov and H. Larochelle, "Efficient learning of deep boltzmann machines," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, JMLR Workshop and Conference Proceedings, 2010, pp. 693–700.

[26] K. H. Cho, T. Raiko, and A. Ilin, "Gaussian-bernoulli deep boltzmann machine," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2013, pp. 1–7.

[27] H. Poon and P. Domingos, "Sum-product networks: A new deep architecture," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, IEEE, 2011, pp. 689–690.

[28] W. M. van der Aalst, "Responsible data science: Using event data in a "people friendly" manner," in *International Conference on Enterprise Information Systems*, Springer, 2016, pp. 3–28.

[29] M. Iman, F. C. Delicato, C. M. De Farias, L. Pirmez, I. L. Dos Santos, and P. F. Pires, "Theseus: A routing system for shared sensor networks," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, IEEE, 2015, pp. 108–115.

[30] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[31] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in neural information processing systems*, vol. 1, 1988.

[32] E. Hines and R. Hutchinson, "Application of multi-layer perceptrons to facial feature location," in *Third International Conference on Image Processing and its Applications, 1989.*, IET, 1989, pp. 39–43.

[33] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]," *IEEE Transactions on Neural Networks*, vol. 20, no. 3, pp. 542–542, 2009.

[34] K. Johnson, "Alexa scientists reduce speech recognition errors up to 22% with semi-supervised learning," *Ventur. Beat*, 2019.

[35] T. Dietterich, "Overfitting and undercomputing in machine learning," *ACM computing surveys (CSUR)*, vol. 27, no. 3, pp. 326–327, 1995.

[36] T. Poggio, K. Kawaguchi, Q. Liao, *et al.*, "Theory of deep learning iii: The non-overfitting puzzle," *CBMM Memo*, vol. 73, 2018.

[37] J. D. Rodriguez, A. Perez, and J. A. Lozano, "Sensitivity analysis of k-fold cross validation in prediction error estimation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 3, pp. 569–575, 2009.

[38] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *2014 science and information conference*, IEEE, 2014, pp. 372–378.

[39] T. Clarke and T. Ronayne, "Categorial approach to machine learning," in *Conference Proceedings 1991 IEEE International Conference on Systems, Man, and Cybernetics*, 1991, 1563–1568 vol.3. DOI: 10.1109/ICSMC.1991.169911.

[40] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological review*, vol. 65, no. 6, p. 386, 1958.

[41] D. Anderson and G. McNeill, "Artificial neural networks technology," *Kaman Sciences Corporation*, vol. 258, no. 6, pp. 1–83, 1992.

[42] M. Minsky and S. Papert, "An introduction to computational geometry," *Cambridge tiass., HIT*, vol. 479, p. 480, 1969.

[43] B. Karlik and A. V. Olgac, "Performance analysis of various activation functions in generalized mlp architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems*, vol. 1, no. 4, pp. 111–122, 2011.

[44] P. Sibi, S. A. Jones, and P. Siddarth, "Analysis of different activation functions using back propagation neural networks," *Journal of theoretical and applied information technology*, vol. 47, no. 3, pp. 1264–1268, 2013.

[45] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.

[46] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[47] M. W. Gardner and S. Dorling, "Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences," *Atmospheric environment*, vol. 32, no. 14-15, pp. 2627–2636, 1998.

[48] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization.," *Journal of machine learning research*, vol. 13, no. 2, 2012.

[49] D. Maclaurin, D. Duvenaud, and R. Adams, "Gradient-based hyperparameter optimization through reversible learning," in *International conference on machine learning*, PMLR, 2015, pp. 2113–2122.

[50] J. W. Tweedale, "An application of transfer learning for maritime vision processing using machine learning," in *International Conference on Intelligent Decision Technologies*, Springer, 2018, pp. 87–97.

[51] D. Cireşan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *The 2011 international joint conference on neural networks*, IEEE, 2011, pp. 1918–1921.

[52] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE transactions on neural networks*, vol. 8, no. 1, pp. 98–113, 1997.

[53] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," *arXiv preprint arXiv:1506.06579*, 2015.

[54] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[55] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[56] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[57] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[58] T. Zubair, A. Raquib, and J. Qadir, "Combating fake news, misinformation, and machine learning generated fakes: Insight's from the islamic ethical tradition," *ICR Journal*, vol. 10, no. 2, pp. 189–212, 2019.

[59] K. Sato, C. Young, and D. Patterson, "An in-depth look at google's first tensor processing unit (tpu)," *Google Cloud Big Data and Machine Learning Blog*, vol. 12, 2017.

[60] H. S. Chang, M. C. Fu, J. Hu, and S. I. Marcus, "Google deepmind's alphago: Operations research's unheralded role in the path-breaking achievement.," *OR/MS Today*, vol. 43, no. 5, pp. 24–30, 2016.

[61] L. Y. Pratt, *Transferring previously learned backpropagation neural networks to new learning tasks*. Rutgers The State University of New Jersey-New Brunswick, 1993.

[62] F. Zhuang, Z. Qi, K. Duan, *et al.*, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[63] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, *et al.*, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[64] S. Zebang and K. Sei-ichiro, "Densely connected autoencoders for image compression," in *Proceedings of the 2nd International Conference on Image and Graphics Processing*, 2019, pp. 78–83.

[65] D. A. Ferrucci, "Introduction to "this is watson"," *IBM Journal of Research and Development*, vol. 56, no. 3.4, pp. 1–1, 2012.

[66] W.-D. J. Zhu, B. Foyle, D. Gagné, *et al.*, *IBM Watson content analytics: discovering actionable insight from your content*. IBM Redbooks, 2014.

[67] N. Bostrom and E. Yudkowsky, "The ethics of artificial intelligence," in *Artificial intelligence safety and security*, Chapman and Hall/CRC, 2018, pp. 57–69.

[68] A. Etzioni and O. Etzioni, "Incorporating ethics into artificial intelligence," *The Journal of Ethics*, vol. 21, no. 4, pp. 403–418, 2017.

[69] A. Jobin, M. Ienca, and E. Vayena, "The global landscape of ai ethics guidelines," *Nature Machine Intelligence*, vol. 1, Sep. 2019. DOI: 10.1038/s42256-019-0088-2.

[70] A. Gkoulalas-Divanis, G. Loukides, and J. Sun, "Publishing data from electronic health records while preserving privacy: A survey of algorithms," *Journal of biomedical informatics*, vol. 50, pp. 4–19, 2014.

[71] F. Zhuang, Z. Qi, K. Duan, *et al.*, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[72] S. Voghoei, N. H. Tonekaboni, J. G. Wallace, and H. R. Arabnia, "Deep learning at the edge," in *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, IEEE, 2018, pp. 895–901.

[73] N. N. Das, N. Kumar, M. Kaur, V. Kumar, and D. Singh, "Automated deep transfer learning-based approach for detection of covid-19 infection in chest x-rays," *Irbm*, 2020.

[74] A. Jaiswal, N. Gianchandani, D. Singh, V. Kumar, and M. Kaur, "Classification of the covid-19 infected patients using densenet201 based deep transfer learning," *Journal of Biomolecular Structure and Dynamics*, vol. 39, no. 15, pp. 5682–5689, 2021.

[75] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, *Adv. neural inf. process. syst.* 2014.

[76] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *International conference on artificial neural networks*, Springer, 2018, pp. 270–279.

[77] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, *et al.*, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.

[78] H. Ravishankar, P. Sudhakar, R. Venkataramani, *et al.*, "Understanding the mechanisms of deep transfer learning for medical images," in *Deep learning and data labeling for medical applications*, Springer, 2016, pp. 188–196.

[79] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering–a systematic literature review," *Information and software technology*, vol. 51, no. 1, pp. 7–15, 2009.

[80] L. Wan, R. Liu, L. Sun, H. Nie, and X. Wang, "Uav swarm based radar signal sorting via multi-source data fusion: A deep transfer learning framework," *Information Fusion*, vol. 78, pp. 90–101, 2022.

[81] A. Albayrak, "Classification of analyzable metaphase images using transfer learning and fine tuning," *Medical & Biological Engineering & Computing*, vol. 60, no. 1, pp. 239–248, 2022.

[82] S. Kumar *et al.*, "Mcft-cnn: Malware classification with fine-tune convolution neural networks using traditional and transfer learning in internet of things," *Future Generation Computer Systems*, vol. 125, pp. 334–351, 2021.

[83] Y. Wang, Z. Feng, L. Song, X. Liu, and S. Liu, "Multiclassification of endoscopic colonoscopy images based on deep transfer learning," *Computational and Mathematical Methods in Medicine*, vol. 2021, 2021.

[84] M. Akhand, S. Roy, N. Siddique, M. A. S. Kamal, and T. Shimamura, "Facial emotion recognition using transfer learning in the deep cnn," *Electronics*, vol. 10, no. 9, p. 1036, 2021.

[85] D. Jha, K. Choudhary, F. Tavazza, *et al.*, "Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning," *Nature communications*, vol. 10, no. 1, pp. 1–12, 2019.

[86] M. Talo, U. B. Baloglu, Ö. Yıldırım, and U. R. Acharya, "Application of deep transfer learning for automated brain abnormality classification using mr images," *Cognitive Systems Research*, vol. 54, pp. 176–188, 2019.

[87] Z. Wu, H. Jiang, K. Zhao, and X. Li, "An adaptive deep transfer learning method for bearing fault diagnosis," *Measurement*, vol. 151, p. 107 227, 2020.

[88] W. Mao, L. Ding, S. Tian, and X. Liang, "Online detection for bearing incipient fault based on deep transfer learning," *Measurement*, vol. 152, p. 107 278, 2020.

[89] H. Phan, O. Y. Chén, P. Koch, *et al.*, "Towards more accurate automatic sleep staging via deep transfer learning," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 6, pp. 1787–1798, 2020.

[90] P. Perera and V. M. Patel, "Deep transfer learning for multiple class novelty detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 544–11 552.

[91] Y. Xu, Y. Sun, X. Liu, and Y. Zheng, "A digital-twin-assisted fault diagnosis using deep transfer learning," *Ieee Access*, vol. 7, pp. 19 990–19 999, 2019.

[92] K. Han, A. Vedaldi, and A. Zisserman, "Learning to discover novel visual categories via deep transfer clustering," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8401–8409.

[93] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 79–88.

[94] M. Sabatelli, M. Kestemont, W. Daelemans, and P. Geurts, "Deep transfer learning for art classification problems," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018.

[95] D. George, H. Shen, and E. Huerta, "Deep transfer learning: A new deep learning glitch classification method for advanced ligo," *arXiv preprint arXiv:1706.07446*, 2017.

[96] R. Ding, X. Li, L. Nie, *et al.*, "Empirical study and improvement on deep transfer learning for human activity recognition," *Sensors*, vol. 19, no. 1, p. 57, 2018.

[97] M. Zeng, M. Li, Z. Fei, Y. Yu, Y. Pan, and J. Wang, "Automatic icd-9 coding via deep transfer learning," *Neurocomputing*, vol. 324, pp. 43–50, 2019.

[98] H. Kaya, F. Gürpınar, and A. A. Salah, "Video-based emotion recognition in the wild using deep transfer learning and score fusion," *Image and Vision Computing*, vol. 65, pp. 66–75, 2017.

[99] B. Ay, B. Tasar, Z. Utlu, K. Ay, and G. Aydin, "Deep transfer learning-based visual classification of pressure injuries stages," *Neural Computing and Applications*, pp. 1–12, 2022.

[100] P. Li, H. Cui, A. Khan, *et al.*, "Deep transfer learning for wifi localization," in *2021 IEEE Radar Conference (RadarConf21)*, IEEE, 2021, pp. 1–5.

[101] Y. Celik, M. Talo, O. Yildirim, M. Karabatak, and U. R. Acharya, "Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images," *Pattern Recognition Letters*, vol. 133, pp. 232–239, 2020.

[102] C. Liu, Z. Wei, D. W. K. Ng, J. Yuan, and Y.-C. Liang, "Deep transfer learning for signal detection in ambient backscatter communications," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1624–1638, 2020.

[103] S. Deepak and P. Ameer, "Brain tumor classification using deep cnn features via transfer learning," *Computers in biology and medicine*, vol. 111, p. 103 345, 2019.

[104] R. Mormont, P. Geurts, and R. Marée, "Comparison of deep transfer learning strategies for digital pathology," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2018, pp. 2262–2271.

[105] Z. Yang, W. Yu, P. Liang, *et al.*, "Deep transfer learning for military object recognition under small training set condition," *Neural Computing and Applications*, vol. 31, no. 10, pp. 6469–6478, 2019.

[106] Y. Gao and K. M. Mosalam, "Deep transfer learning for image-based structural damage recognition," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 9, pp. 748–768, 2018.

[107] Y. Yu, H. Lin, J. Meng, X. Wei, H. Guo, and Z. Zhao, "Deep transfer learning for modality classification of medical images," *Information*, vol. 8, no. 3, p. 91, 2017.

[108] S. Wang, Z. Li, Y. Yu, and J. Xu, "Folding membrane proteins by deep transfer learning," *Cell systems*, vol. 5, no. 3, pp. 202–211, 2017.

[109] D. Joshi, V. Mishra, H. Srivastav, and D. Goel, "Progressive transfer learning approach for identifying the leaf type by optimizing network parameters," *Neural Processing Letters*, vol. 53, no. 5, pp. 3653–3676, 2021.

[110] A. Moeed, G. Hagerer, S. Dugar, *et al.*, "An evaluation of progressive neural networksfor transfer learning in natural language processing," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 1376–1381.

[111] Y. Gu, Z. Ge, C. P. Bonnington, and J. Zhou, "Progressive transfer learning and adversarial domain adaptation for cross-domain skin disease classification," *IEEE journal of biomedical and health informatics*, vol. 24, no. 5, pp. 1379–1393, 2019.

[112] J. Gideon, S. Khorram, Z. Aldeneh, D. Dimitriadis, and E. M. Provost, "Progressive neural networks for transfer learning in emotion recognition," *arXiv preprint arXiv:1706.03256*, 2017.

[113] M. Loey, G. Manogaran, and N. E. M. Khalifa, "A deep transfer learning model with classical data augmentation and cgan to detect covid-19 from chest ct radiography digital images," *Neural Computing and Applications*, pp. 1–13, 2020.

[114] X. Li, W. Zhang, Q. Ding, and X. Li, "Diagnosing rotating machines with weakly supervised data using deep transfer learning," *IEEE transactions on industrial informatics*, vol. 16, no. 3, pp. 1688–1697, 2019.

[115] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Transactions on systems, man, and cybernetics: systems*, vol. 49, no. 1, pp. 136–144, 2017.

[116] M. Simon, E. Rodner, and J. Denzler, "Imagenet pre-trained models with batch normalization," *arXiv preprint arXiv:1612.01452*, 2016.

[117] B. Neyshabur, H. Sedghi, and C. Zhang, "What is being transferred in transfer learning?" *Advances in neural information processing systems*, vol. 33, pp. 512–523, 2020.

[118] T. Mensink, J. Uijlings, A. Kuznetsova, M. Gygli, and V. Ferrari, "Factors of influence for transfer learning across diverse appearance domains and task types," *arXiv preprint arXiv:2103.13318*, 2021.

[119] A. Albayrak, "Classification of analyzable metaphase images using transfer learning and fine tuning," *Medical & Biological Engineering & Computing*, vol. 60, no. 1, pp. 239–248, 2022.

[120] N. N. Das, N. Kumar, M. Kaur, V. Kumar, and D. Singh, "Automated deep transfer learning-based approach for detection of covid-19 infection in chest x-rays," *Irbm*, 2020.

[121] A. Jaiswal, N. Gianchandani, D. Singh, V. Kumar, and M. Kaur, "Classification of the covid-19 infected patients using densenet201 based deep transfer learning," *Journal of Biomolecular Structure and Dynamics*, vol. 39, no. 15, pp. 5682–5689, 2021.

[122] D. Jha, K. Choudhary, F. Tavazza, *et al.*, "Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning," *Nature communications*, vol. 10, no. 1, pp. 1–12, 2019.

[123] M. Simon, E. Rodner, and J. Denzler, "Imagenet pre-trained models with batch normalization," *arXiv preprint arXiv:1612.01452*, 2016.

[124] B. Ay, B. Tasar, Z. Utlu, K. Ay, and G. Aydin, "Deep transfer learning-based visual classification of pressure injuries stages," *Neural Computing and Applications*, pp. 1–12, 2022.

[125] A. Ashraf, S. Naz, S. H. Shirazi, I. Razzak, and M. Parsad, "Deep transfer learning for alzheimer neurological disorder detection," *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30 117–30 142, 2021.

[126] Y. Celik, M. Talo, O. Yildirim, M. Karabatak, and U. R. Acharya, "Automated invasive ductal carcinoma detection based using deep transfer learning with whole-slide images," *Pattern Recognition Letters*, vol. 133, pp. 232–239, 2020.

[127] C. Liu, Z. Wei, D. W. K. Ng, J. Yuan, and Y.-C. Liang, "Deep transfer learning for signal detection in ambient backscatter communications," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 1624–1638, 2020.

[128] S. Deepak and P. Ameer, "Brain tumor classification using deep cnn features via transfer learning," *Computers in biology and medicine*, vol. 111, p. 103 345, 2019.

[129] Y.-X. Wang, D. Ramanan, and M. Hebert, "Growing a brain: Fine-tuning by increasing model capacity," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2471–2480.

[130] T. Takase, S. Oyama, and M. Kurihara, "Effective neural network training with adaptive learning rate based on training loss," *Neural Networks*, vol. 101, pp. 68–78, 2018.

[131] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[132] H. Iiduka, "Appropriate learning rates of adaptive learning rate optimization algorithms for training deep neural networks," *IEEE Transactions on Cybernetics*, 2021.

[133] A. Koul, S. Ganju, and M. Kasam, *Practical Deep Learning for Cloud, Mobile, and Edge: Real-World AI & Computer-Vision Projects Using Python, Keras & TensorFlow*. O'Reilly Media, 2019.

[134] M. Iman, *Expanse evaluation codes*, `https://github.com/mrezaim/Expanse`, [Accessed: 2022].

[135] Y. LeCun, C. Cortes, and C. J. Burges, *The mnist database of handwritten digits*, `http://yann.lecun.com/exdb/mnist/`, [Accessed: 2022].

[136] S. Arora and M. S. Bhatia, "Handwriting recognition using deep learning in keras," in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, IEEE, 2018, pp. 142–145.

[137] D. C. Cireşan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Deep, big, simple neural nets for handwritten digit recognition," *Neural computation*, vol. 22, no. 12, pp. 3207–3220, 2010.

[138] M. R. Shamsuddin, S. Abdul-Rahman, and A. Mohamed, "Exploratory analysis of mnist handwritten digit for machine learning modelling," in *International Conference on Soft Computing in Data Science*, Springer, 2018, pp. 134–145.

[139] J. Pomerat, A. Segev, and R. Datta, "On neural network activation functions and optimizers in relation to polynomial regression," in *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 6183–6185.

[140] P. Ghosh, A. A. Anjum, A. Karim, *et al.*, "A comparative study of different deep learning model for recognition of handwriting digits," *International conference on iot based control networks and intelligent systems (ICICNIS 2020)*, pp. 857–866, 2021.

[141] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.