A Survey of Modern Transfer Learning and Advanced Intelligent Systems

Shaun Pritchard, Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton FL, USA, spritchard2021@fau.edu

***Abstract* - Transfer learning is the process by which systems can quickly adapt themselves to new situations, new tasks, and new environments. When there is only a limited amount of data available in the target domain, machine learning systems can use auxiliary data and models to help solve the target problem. This paper presents a comprehensive analysis of transfer-learning methods, including narrow and deep approaches in the fields of Natural Language Processing (NLP) and Computer Vision (CV). Throughout this paper, we explore the objectives of intelligent systems that motivated the development of transfer learning in its many forms in contrast to the advancements and problems being solved with current technology. This paper seeks to explore the important practical issues in the field of machine learning and AI. Also, in comparing a comprehensive overview from baseline research and other sources on the topic[1].**

Keywords – Transfer Learning, Advances in Transfer Learning, Homogeneous Transfer Learning, Deep Learning, Data Mining, Machine Learning

Introduction

The advancements in supervised and unsupervised machine learning have led to remarkable advances in artificial intelligence over the last few years. Currently, we can build autonomous vehicles, intelligent robots, even cancer detection systems, and smart sentiment web bots that perform on par with or even better than humans and have high accuracy. Machine learning methods such as transfer learning allow a model developed for one task to be reused to develop a model for another. Transfer learning is the process by which an algorithm extracts knowledge from several application scenarios to improve learning performance in a target scenario. As mentioned in the baseline[1], Increasingly, machine learning is being integrated into real-world applications because of its success and benefits. According to traditional machine learning, both training and testing data are drawn from the same feature space and are distributed in the same way[2]. In practice, however, this assumption may be false. In many ways, transfer learning and machine learning are inspired by biological learning systems. Even today, there are special fields devoted to Evolutionary Computation (EC) approaches that, in their own right, are inspired by nature and process optimization problems in a stochastic manner. Transfer Learning is a notion that is similar to machine learning transfer in the field of education and learning psychology. This refers to the process by which past experiences acquired from previous source tasks can be used to influence future learning and performance in a target situation. [3] Even though machine learning has advanced and continues to advance, further improvements are still possible because humans can learn much more efficiently than current ML methods. Humans can learn certain tasks from only a few examples, apply their knowledge to conditions they have never faced before, and continuously adapt and upgrade their skills so they can perform a wide range of tasks and problems. Humans can identify similarities across different learning problems through knowledge acquired from past experiences. Through collaboration and shared expertise, human beings can also learn from each other.

It is unnecessary to learn everything from scratch when you can build on the experiences of others. As a result of these abilities of humans, as well as the fact that the performance of single-task ML techniques is reaching theoretical learning limits, the focus of ML research has shifted from learning a single task in isolation to determining how knowledge can be transferred across domains, tasks, and agents that are related to each other[2]. In this paper, a comprehensive review of transfer learning and its advancements in applications is provided; This paper is divided into the following sections: *Transfer learning, Homogeneous transfer learning, Heterogeneous transfer learning, Issues for Transfer Learning to Solve, Advances and Transfer Learning Algorithms, Baseline Comparison, Critiques of Transfer learning, Conclusion, Future Works*

1) **Transfer Learning:**

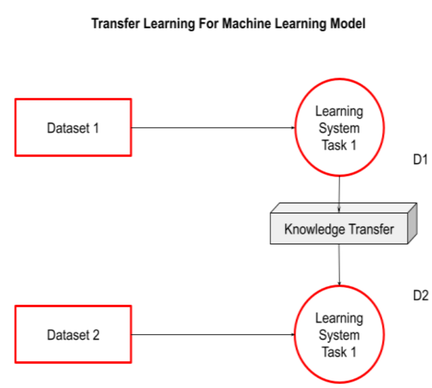
A variety of terms have been used to describe transfer learning in AI. According to the similarity between domains and the availability of labeled and unlabeled data, there are three types of transfer learning problems: inductive transfer learning, transductive transfer learning, and unsupervised transfer learning. The process of transduction consists of converting one kind of experience (training) into another kind (test).

By contrast, induction is the application of general rules based on observable training cases to test cases. A recent body of research in this field has, however, broadened the scope of transfer learning due to advances in deep learning, and a new and more flexible taxonomy has emerged. Transfer learning problems are generally classified into two classes based on the similarity of the domain, regardless of whether data is labeled or unlabeled, homogeneous transfer learning and heterogeneous transfer learning. Each with various algorithmic methods that will be discussed later in their research. Just know that transferring learning is primarily implemented in the ML sub-fields of Natural Language Processing (NLP), Computer Vision (C), and Reinforcement Learning (RL). Transfer learning is a machine learning method in which a model developed to share knowledge for one task is used as the basis for another model. In a knowledge transfer scenario, the problem from which knowledge is gained is called the source problem, and the problem from which that knowledge is transferred is called the target problem. We can recognize the models of knowledge or data transfer as domains and the data being transferred as the task[4].

If you compare it to when you were a child (domain) and learned that the stove was hot, the knowledge of that danger was stored away until you were an adult (different domain) and could infer that when cooking there may be a risk of danger that helped guide your decisions in the kitchen. In general, the distance between domains can be measured by analyzing features used to describe the data. Pixels or patches in an image pattern can be considered features in image analysis, such as color or shape. Words and phrases can also be considered features in NLP. The more we know about a domain's proximity to another domain, the easier it is to propagate AI models from the well-developed domain to the less-developed domain, which will make AI less data-dependent. The success of transfer learning applications can be determined by these factors. As mentioned previously, this analogy is a simplified interpretation of transfer learning. Here is a deeper look at the algorithm mathematics abscess for the transfer learning process. By following the notations introduced [2], we define "domain," "task," and "transfer learning." as follows:

Domain: A domain $\mathfrak{D} = \{X, P(X)\}$ is defined by two components:
A feature space X and a discrete probability distribution space $P(X)$ where $X = \{x_1, x_2, x_3, \ldots, x_n\} \in X$
A difference between two domains is either a difference between their feature spaces **(Xt or Xs),** or a difference in the marginal distributions **(P(Xt)** or **P(Xs)).**

Task: Given a specific domain $\mathfrak{D}$, a task $\mathcal{T}=\{Y, f(.)\}$ consists of two parts:
A label space Y and a predictive function $f(.)$, which is not observed but can be learned from some training data $\{(x_i, y_i) | I \in \{1, 2, 3, \ldots, N\}$, where $x_i \in X$ and $y_i \in Y$ }. From a stochastic perspective $f(x_i)$, can also be written as $p(y_i|x_i)$, so we can rewrite task $\mathcal{T}$ as $\mathcal{T}=\{Y, P(Y|X)\}$. In general, if two tasks are different, then they may have different label spaces($Yt \neq Ys$) or different conditional probability distributions ($P(Yt|Xt) \neq P(Ys|Xs)$). Given a source domain $\mathfrak{D}s$ and corresponding learning task $\mathcal{T}s$, a target domain $\mathfrak{D}t$ and learning task $\mathcal{T}t$, transfer learning aims to improve the learning of the conditional probability distribution $P(Yt|Xt)$ in $\mathfrak{D}t$ with the information gained from $\mathfrak{D}s$ and $\mathcal{T}s$, where $\mathfrak{D}t \neq \mathfrak{D}s$ or $\mathcal{T}t \neq \mathcal{T}s$. $\mathfrak{D}t \neq \mathfrak{D}s$ or $\mathcal{T}t \neq \mathcal{T}s$, which results in four common transfer learning that defines domain and task. The example below shows how different tasks are transferred to different domains sharing knowledge of previously training models.



2) Transfer Learning Applications

A wide range of applications is increasingly being applied to machine learning techniques. Though still a very young field of study with potential for advancement. As a robust technique, transfer learning is valuable for many applications. Following is a list of applications in which transfer learning can be used as follows:

1. Transfer Learning in Computer Vision - utilizing Transfer Learning in CV implements object detection, image analysis, and general visual tasks such as image classification and video classification[5].
2. Privacy-Preservations Transfer Learning - Transfer Learning is being used for applications in social networking, banking, supply chain management, and health care. TL is used to protect sensitive information such as medical records, financial transactions, and personal information inside various files.
3. Transfer Learning in NLP - Many NLP tasks involve transfer learning, especially when models must be trained with limited data. Among NLP's pillars are sentiment analysis, TTS, STT semantics, and many other fascists of natural language and text and audio[6].
4. Transfer Learning in Dialogue Systems - There are two types of dialogue systems: open-domain dialogue systems and task-oriented dialogue systems[7].
5. Transfer Learning in Bioinformatics - The field of bioinformatics is interdisciplinary and covers diverse fields with areas such as biology, biochemistry, machine learning, data management, information retrieval, and computer science, among others. Technology in biology brings rapid growth and a large amount of data[8]–[11] .
6. Transfer Learning in In Recommender Systems - Many intelligent systems rely on recommendation systems. Netflix and other streaming apps are examples of recommender systems. With this technology, TL is setting a new standard[12].
7. Transfer Learning in Activity Recognition - Human behavior recognition from sensor observations. Typically, from smart, tracking, mobile devices, and wearable devices[13].
8. Transfer Learning in Urban Computing - Currently, smartphones, vehicles, and infrastructure (e.g. Edge-network device sensors, traffic cameras, air-quality monitoring stations, GPS points, online posts, road conditions, and weather conditions) continuously generate information about with sensory data is being utilized with TL[13], [14].

Current ML technology has a lot of opportunities to improve in these applications, leading back to primary issues with resources, data, and other factors affecting data and computation. Although this paper is a survey on transfer learning, the breadth of reviewing the application should stand on its own since it is quite a broad topic. Hence, we will only review the major aspects of the algorithms and functionality here.

3) Homogeneous Transfer Learning

homogeneous transfer learning $Xt = Xs$ and $Y\mathrm{t} = Ys$. Therefore, we want to bridge the gap in the data distributions between the source and target domains, i.e. address $\mathrm{P}(\mathrm{Xt}) \neq \mathrm{P}(\mathrm{Xs})$ and/or $\mathrm{P}(\mathrm{Yt}|\mathrm{Xt}) \neq \mathrm{P}(\mathrm{Ys}|\mathrm{Xs})$. The solutions to homogeneous transfer learning problems use one of the following general strategies: Trying to correct for the marginal distribution differences in the source and target ($\mathrm{P}(\mathrm{Xt}) \neq \mathrm{P}(\mathrm{Xs})$). Trying to correct for the conditional distribution difference in the source and target ($\mathrm{P}(\mathrm{Yt}|\mathrm{Xt}) \neq \mathrm{P}(\mathrm{Ys}|\mathrm{Xs})$). The correction should be made for both marginal and conditional distribution differences in the source and target. According to the baseline research, there are five types of homogeneous transfer learning solutions: instance-based, feature-based *(symmetric or asymmetric)*, model-parameter-based, relational-information-based, and hybrid-based approaches[1]. We will explore these algorithms in more depth. Homogeneous transfer learning supports several techniques that are straightforward implementations of transferring learned data between domains. These are some of the techniques that fall under the realm of homogeneous transfer learning.

Instance-based transfer learning

The goal of instance-based transfer learning is to repurpose labeled data from the source domain for training a more accurate model for the target domain. If two domains are quite similar, data from one domain can be combined with data from another domain. Then it becomes a standard machine learning problem within a single domain. However, in many cases, the "direct adoption" of source domain instances does not solve the target problem. Instance-based approaches differ in their weighting strategies. Bias-variance analysis to understand this motivation, When feature space $PX s = PX t \textit{ but } PY /X_s = PY /X_t$, we refer to the problem setting as non-inductive transfer learning. Were inductive transferring is, $PY |X_s = PY |X_t$[15]

Feature-based transfer learning

They can be applied to both homogeneous and heterogeneous problems. In heterogeneous problems, these methods are mainly aimed at reducing the gap between the feature spaces of the source and target domains, while in homogeneous problems, they aim at reducing the gap between the marginal and conditional distributions of the source and target domains. There are two types of feature-based transfer learning approaches[15].

- *Asymmetric Feature Transformation:* This approach employs a transformation of ($\Phi s/\Phi t$) to transform one domain into the other domain. When the source and target domains share a common label space and can be transformed without biasing context features, this method is most effective.
- *Symmetric Feature Transformation:* By transforming two domains into a common latent feature space, underlying structures can be identified with the aid of predictability, while the marginal distribution between the domains is reduced.

Parameter-based (Model) transfer learning

The aim of this category of transfer learning is to transfer knowledge by using shared parameters between the source and target domain learners. The learned knowledge can also be transferred by creating multiple source learning models and combining the reweighted learning models (ensemble learners) to create an improved target learning model. Parameter-based methods have the advantage that a well-trained model on the source domain has learned a structure, which can be used as a transfer factor when two tasks are related. Deep neural networks are trained with weights that are initially close to zero, and as more training samples are accumulated, the weights change. Nevertheless, training a deep model this way requires a great deal of time and effort to collect and label data. It is therefore advantageous to start with previously trained weights from another domain and then fine-tune them for a new domain It is therefore advantageous to start with previously trained weights from another domain and then fine-tune them for a new domain[14] .

Relational-based transfer learning

This category of transfer learning involves learning the common relationships between the source domain and the target domain. As opposed to instance-based, feature-based and model-based transfer learning methods, each assuming that data instances are independent and identically distributed. This method incorporates first- and second-order relations based on the assumption that at least two related relationship domains share some similar relation-independent structural regularities that can be extracted from the source domain [15].

Hybrid-based transfer learning

This category focuses on transferring knowledge through both instances and shared parameters. This is a relatively new approach and supports both instances and parameters [2].

5) Heterogeneous transfer learning

In heterogeneous transfer learning, the source and target have different feature spaces $Xt \neq Xs$ (generally non-overlapping) and/or $Yt \neq Ys$, as the source and target domains may share no features and/or labels. Consequently, heterogeneous transfer learning solutions eliminate the feature space gap and reduce the problem to one of homogeneous transfer learning. A number of advanced transfer learning methodologies are also available. Including more advanced state-of-art methods such as both homogeneous and heterogeneous transfer learning, Adversarial transfer Learning, Transfer Learning in Reinforcement Learning, Multi-task Learning, AutoTL, Transitive Transfer Learning, Few Shot, and many others discussed in baseline research that will be discussed later[16]. The ultimate objective of transfer learning is to enhance the performance of a target task using an auxiliary/source domain feature space. As a result, transferring knowledge from one domain to another could harm the target model. Heterogeneous transfer learning solutions aim to solve this issue through methods that require far fewer target labels on data. In transfer learning, three fundamental questions predominate: when to transfer, how to transfer, and what to transfer.

6) Issues for Transfer Learning to Solve

The goal of transfer learning is to improve the quality and speed of the currently available machine learning algorithm by overcoming labeled data, sparsity, imbalance problems, scarcity, avoiding redundant learning and model retraining, and efficiently utilizing computational power resources, as well as using less labeled data more effectively. The use of deep neural networks in machine learning has become so common that it has become so expensive to retrain complex models with a large number of parameters [17] . For many real-world problems, it is not practical to label millions of data points, so transfer learning can be very useful. Accordingly, the success of machine learning will be driven by transfer learning. In order to succeed, machine learning needs a large amount of labeled data. Many applications have only small datasets. Nevertheless, high-quality labels are scarce. Traditional machine learning methods fail to perform well when they're applied to new tasks due to overfitting. Experience and personalization are important factors. It is imperative in our modern civilization to provide personalized services based on the needs of users. Retrospectively, we can only collect a very small amount of personal information from individual users in many real-world applications. Consequently, traditional machine learning methods have a scarcity problem when trying to adopt a general model to a specific situation. Traditionally, machine learning assumes that training and test data come from the same distribution. Machine learning models need to be robust. In many practical scenarios, this assumption is not feasible. In many cases, the distribution varies over time and space, as well as from situation to situation, which means we may never have new training data to go with the same test distribution. Before they can be used in situations that differ from the training data, trained models need to be adapted.

The privacy and security of our users are important issues, and we often need to leverage multiple data sets in our applications to work with other organizations. In many cases, these data sets belong to different parties and cannot be shared for reasons of privacy or security. By transferring the key features of each data set and adapting them to build a new model, transfer learning can bridge the gap by allowing resources of metadata to be modeled together safely and securely. Thus, the privacy of the scarce data can be assured through the adaptation of the general model. Artificial intelligence can be promoted in geographically and technologically underdeveloped areas through transfer learning. The ability to transfer knowledge from one domain to another allows machine learning systems to extend their range of applicability beyond their original creation. This generalization ability helps make AI more accessible and more robust in many areas where AI talents or resources such as computing power, data, and hardware might be scarce[4], [10].

7) Advances and Transfer Learning Algorithms

Although the baseline research focuses on heterogeneous transfer learning systems specifically for NLP and image-processing, this paper also discusses some advances in comparison and contrast to the baseline research[1] while deliberating on some newer more advanced techniques. The algorithms and methods we will review are Heterogeneous transfer learning systems applicable to various applications as resulted in the baseline paper for NLP[16]. The following will show the implementations of the research as well as further research into more robust models that result in optimal robust AI models. Due to brevity, I will only mention a few key advancements with explanations and mentions of Homogeneous transfer learning methods as follows:[1], [9], [18].

Adversarial Transfer Learning

In machine learning, generative modeling can be used to apply transfer learning. This results in adversarial models. Unsupervised generative modeling is one method to reduce the dependence on labeled data. A target domain may have limited labeled data, but a source domain may have abundant unlabeled data[18]. There are two types of generative models, explicit and implicit models. Among generative models are generative adversarial networks (GANs)[19]. The GAN framework. It is based on the competition between two sub-networks, a generator, and a discriminator. A vector sampled from a prior distribution is mapped to the data space by the generator. The discriminator attempts to distinguish true samples of data from generated samples, while the generator attempts to trick the discriminator. Another type of adversarial transfer learning model takes a feature-based transfer learning approach, which finds a common feature space with an adversarial objective[19], [20].

Transfer Learning in Reinforcement Learning

As defined in the Markov decision process(MDP), reinforcement learning defines an actor in the environment who changes state based on the cumulative past actions. The purpose of transfer learning is to enhance the performance of a target-domain MDP by leveraging the knowledge and experience of one or more related but different source MDPs. Without loss of generality, we mainly discuss the case with a single source domain in this section for clarity. The domain of an MDP M, that is, DM, includes the state space S and the action space A[21]. The domain of a continuous MDP mainly represents the continuous state variables and action space. The state-space or the action space of two MDPs that belong to different domains will differ. When some MDPs have different domains, transfer learning is dependent on the interdomain mapping between the source and target domains. Considering a domain M, the task describes the remaining components of an MDP, such as the transition function PM and the reward function RM. Different MDPs have different reward functions or dynamics. PM and RM can be unknown to the agent and require exploitation and exploration. n MDP M is defined as a tuple $< SM , AM ,PM ,RM ,\gamma >$. State and action spaces can be infinite, and SM and AM denote these spaces. For continuous states, SM ∈ Rd stands for the state variables. An MDP is made up of state and action variables.

By evaluating the transition function PM : SM * AM * SM, the next visited state is determined based on the current state and the actions taken. There are deterministic and stochastic versions of PM. Model-based learning refers to reinforcement learning algorithms that estimate the transition function PM explicitly. RM : SM → RM is the reward function that generates an instantaneous reward whenever a new state is reached. γ denotes a discount factor. It is necessary to interact with the environment to uncover the transition function PM and the reward function R in the majority of reinforcement learning problems[21].

AutoTL: Learning to transfer automatically

Machine learning has become more complex in recent years, driving the trend of automation of machine learning[22]. Human experts are required to perform several tedious steps to achieve an end-to-end machine learning solution. In addition to sample selection, feature engineering, algorithm selection, architecture design, model tuning, and evaluation, several other things can be done. AutoTL stands for automated transfer learning, also known as the automatic transfer learning framework. Learning to Transfer (L2T), which selects transfer learning algorithms automatically through experience. This framework was first proposed by[23]. In a special case of AutoTL, this is a way to identify the best algorithm and model parameters based on the prior transfer learning experiences.

The purpose of the L2T framework is to improve the transferring performance between a source and a target domain. To accomplish this, L2T has two phases. Following a knowledge transfer experience, which is divided into three elements: two source domains, a target domain, and knowledge transfer between them, and improvement of performance, the first phase of the learning process involves learning a reflection function that maps the source and target domains and knowledge transfer to improvement. As an approximation of the performance improvement, the learned reflection function is maximized to determine what should be transferred between two domains during the second phase. When conducting transfer learning several times, an L2T agent keeps a record of the experiences. Each transfer learning experience is expressed as $E_e\ (\langle S_e\ , T_E\ \rangle, a_E\ , l_E\ )$ where $Se\ =\ \{X_e{}^s\ , Y_e{}^s\ \}$and Te = $\{X_e{}^t\ , Y_e{}^t\ \}$ indicates a source domain and a target domain, correspondingly. $X_e{}^*\ \in\ R^{n^*{}_e\, x\, M}$ represents the data matrix, whereas each domain consists of n∗ e examples in an m-dimensional feature space $X_e{}^*$, where the superscript ∗ denotes $\boldsymbol{s}$ or **t** as a source or target domain. $y_e{}^*\ \in$ $Y_e{}^*$denotes an $n_{le}{}^*\ \times\ 1$vector consisting of labels for $X_e{}^*$. Usually the number of labeled examples in the source domain is much larger than that of the target domain, that is, $n_{le}{}^t\ < < n_{le}{}^s$. We consider the setting of homogeneous feature space. heterogeneous label spaces for each pair of domains, that is, $X_e{}^s\ =\ X_e{}^t$ and $Y_e{}^s\ =\ Y_e{}^t\quad ae\ \in\ A\ =\ \{a1,\cdots, aNa\ \}$ denotes a transfer learning algorithm that has been conducted between Se and T

Few Shot Transfer Learning

We want to build models that can generalize from a few samples in few-shot learning. Inspired by humans who can learn a novel concept by observing that a few examples can be enough to learn it. Few, one, and Zero shot learning algorithms are capable of capturing useful information from only a few examples. Observation, mental comprehension, and memory are all part of cognitive processes. A learning system handles testing samples from novel classes not found in the training set. An evaluation method called few-shot learning is one in which the learning algorithm is only presented with one sample from each class. Comparing traditional machine learning to new concepts or labels appearing in test samples, the essential difference is that the new concepts or labels require a "bridge" from prior knowledge to that of the novel classes. A class's semantic properties are its defining characteristics. Thus, instead of learning a mapping from X to Y, where X is an m-dimensional feature space and Y is a label space, we try to learn the following function: X → F where F is a semantic feature space. In addition, we need a knowledge base K, which contains a description of all classes and their semantic features and serves as a bridge between them. We match the semantic features of the example with the features in the knowledge base [24] . To obtain the most similar class to that associated with a data sample, we match the semantic features of the example with the features in the knowledge base [24]. Additionally, I would like to mention Zero-shot learning, which has its origins as a model of rank-based learning that describes new categories at a high level so that the machine can associate them with existing categories it already knows. Machine perception, computer vision, and natural language processing can all benefit from the zero-shot learning method. Zero-shot learning consists essentially of two stages: training and inference. In the training phase, semantic attributes are captured, then at the inference stage, these attributes are used to predict new categories. Nonetheless, these methods solve the problem of making predictions based on a limited or no number of samples to infer the generalization of datasets is textual, audio, or image based[24].

Lifelong Machine learning

Lifelong machine learning is a machine learning system that completes multiple learning tasks T = {T1,T2,T3,⋯} from different domains D = {D1,D2,D3,⋯} over time, and solves the later tasks more effectively with the help of previously solved tasks. A typical lifelong machine learning system uses a knowledge base K B that stores previously learned knowledge learned over time. An event t occurs when the system receives a task t from a corresponding domain t . Based on training data from D_t and knowledge in K B, a typical lifelong machine learning system builds a new model for T_t then extracts the transferable knowledge from (D_t,T_t) to update K B. The updated knowledge base K B is then used to refine the models trained for previous t -1 tasks [25].

Genetic Transfer Learning

Transfer learning (TL) is the process by which some aspects of a machine learning model generated on a source task are transferred to a target task, to simplify the learning required to solve the target. Genetic programming is a technique to create algorithms that can program themselves by simulating biological breeding and Darwinian evolution. Instead of programming a model that can solve a particular problem, genetic programming only provides a general objective and lets the model figure out the details itself. The basic approach is to let the machine automatically test various simple evolutionary algorithms and then "breed" the most successful programs in new generations[26]. While applying the same natural selection, crossover, mutations, and other reproduction approach as evolutionary and genetic algorithms, gene programming takes the process a step further by automatically creating new models and letting the system select its own goals. GP is used to construct feature transformations and an additional learning algorithm is used to fit the final models. Although the method of applying GP to dynamic environments is similar to transfer learning, examining transfer learning has not been addressed in GP The entire process is still an area of active research, TL in Genetic Programming (GP) has not received much attention, since it is normally assumed that an evolved symbolic expression is specifically tailored to a problem's data and thus cannot be used in other problems[27]. experimental results showed that transfer learning methods help GP to achieve better training errors. Importantly, the performance of GP on unseen data when implemented with transfer learning was also considerably improved.

A random initial population (generation 0) of simple programs will be generated based upon basic functions and terminals defined by the human. Each program will be executed, and its fitness measured from the results. The most successful or "fit" programs will be breeding stock to birth a new generation, with some new population members directly copied (reproduction), some through crossover (randomly breeding parts of the programs), and random mutations. Unlike evolutionary programming, an additional architecture-altering operation is chosen, similar to a species, to control the framework of the new generation. These generations are repeated as necessary until the termination criterion is reached and the best program is designated as the result of the run[27], [28].

Critiques of Transfer learning

I feel that Transfer learning is leading to something that will advance the field of machine learning. The context of robust application to derive results based on differential techniques is very exciting. For example, we might combine genetic reinforcement learning with transfer learning. I believe we must explore all aspects of what it means to be human to transfer knowledge from different disciplines as a basis for a human model. In weighing the bias of our decisions on an annual basis, we do not only consider past events but also current ones. Our model considers future actions in the dialect of all machine learning algorithms through probabilistic and stochastic actions. With transfer learning, we can answer more of our questions with fewer resources and with a greater degree of accuracy. The algorithmic process has produced impressive uniformity of data sets that are not correlated, and this is a result of this algorithmic process. It should also be noted that transfer learning is very young and that abstractions for matching data that is not assumed to match require a great deal of trial and error. I believe that the practitioners are going to have to use a combination of these techniques to get the best performance out of the models they develop in the future. While this does not take much trial and error, it is still very optimistic about the plausibility of the ideas it conveys.

8) Baseline comparisons

Baseline research analyzed other heterogeneous transfer learning (HTL) methods used to explore cross-language text categorization and text-to-image characterizations through a variety of algorithms that are imperatively indicative of Surveyed HTL methods that require limited target labels, require limited target labels, and accept unlabeled target data, and methods that require no target labels[1]. A range of heterogeneous transfer learning methods has been developed to address these limitations. The methods were also found to be application-specific, which made their portability to other applications difficult or impossible. Furthermore, the majority of the baseline surveyed methods do not address the difference in label spaces or the difference in marginal/conditional distributions[1]. The Baseline reported that the data provided did not permit a direct comparison between the performance of the surveyed algorithms. As a consequence, there are few similarities among experimental testbeds of the methods in the surveyed categories. Nonetheless, the examination of many of these transfer learning methods yields very high performance. As well, for each unique method to address such knowledge transfer issues, one of the patterns that can be observed is the use of co-occurrence data while other patterns used Canonical Correlation Analysis to solve variant feature space problems. Of these were Symmetric, multi-view ensemble, CCA analysis, SRKDA, determine relatedness of domains using co-occurrence data, multi-source which I found unique in their application\

9) Conclusion

Although there has been tremendous progress in machine learning algorithms, they still need to be improved significantly to replace humans in many applications; a significant amount of academic research remains in the area. Baseline survey methods seek a commonality between the source and target domains so that they can be represented as a single feature space to extract performance-enhancing knowledge without cross-domain noise[1].

Transfer learning addresses these three questions:

- What information in the source is useful and transferrable to the target?
- How can this information be transferred most effectively?
- How can negative information be avoided from being transferred?

As a result of this research, I believe that Transfer Learning can be used to improve data science and AI beyond their current state. There is still a lot of opportunity for an AI practitioner to develop new transfer learning models to fit applicable domains. This is something that intrigues me and is something that I plan to conduct further research on in the future.

Future research should involve extending and comparing the more advanced heterogeneous transfer learning methods described in this paper, using real analyses, and working data. I am also very interested in collaborating with colleagues who are also interested in AI Transfer Learning Methods for Genetic Programming (GP). It appears to be a field of study with great potential to solve data scarcity problems in bioinformatics, for example[29]. This is something I am very passionate about in my research. In addition, I am interested in peer-reviewed studies of robust models and imbalance in transfer data models. Furthermore, I am intrigued by the latest developments found in my research on sentiment analysis and computer vision. The implementation of transfer learning is in its prime, and the scope of application deserves more research into solving today's more complex problems.

10) References

[1] O. Day and T. M. Khoshgoftaar, “A survey on heterogeneous transfer learning,” *Journal of Big Data*, vol. 4, no. 1, pp. 1–42, Dec. 2017, doi: 10.1186/S40537-017-0089-0/FIGURES/5.

[2] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.

[3] “Classics in the History of Psychology -- Thorndike & Woodworth (1901a).” https://psychclassics.yorku.ca/Thorndike/Transfer/transfer1.htm (accessed Apr. 26, 2022).

[4] M. Iman, K. Rasheed, and H. R. Arabnia, "A Review of Deep Transfer Learning and Recent Advancements," Jan. 2022, doi: 10.48550/arxiv.2201.09679.

[5] X. Li *et al.*, "Transfer learning in computer vision tasks: Remember where you come from," *Image and Vision Computing*, vol. 93, p. 103853, Jan. 2020, doi: 10.1016/J.IMAVIS.2019.103853.

[6] S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf, "Transfer Learning in Natural Language Processing," *Proceedings of the 2019 Conference of the North*, pp. 15–18, 2019, doi: 10.18653/V1/N19-5004.

[7] M. Ghazvininejad *et al.*, "A Knowledge-Grounded Neural Conversation Model *", Accessed: Apr. 26, 2022. [Online]. Available: www.aaai.org

[8] T. Wolf, V. Sanh, J. Chaumond, and C. Delangue, "TransferTransfo: A Transfer Learning Approach for Neural Network Based Conversational Agents." [Online]. Available: https://github.com/huggingface/

[9] P. Saxena, A. Maheshwari, and S. Maheshwari, "Predictive modeling of brain tumor: A Deep learning approach," *Advances in Intelligent Systems and Computing*, vol. 1189, pp. 275–285, Nov. 2019, doi: 10.48550/arxiv.1911.02265.

[10] D. Agarwal, G. Marques, I. de la Torre-Díez, M. A. Franco Martin, B. García Zapiraín, and F. Martín Rodríguez, "Transfer learning for alzheimer's disease through neuroimaging biomarkers: A systematic review," *Sensors*, vol. 21, no. 21, Nov. 2021, doi: 10.3390/S21217259.

[11] Z. Ardalan and V. Subbian, "Transfer Learning Approaches for Neuroimaging Analysis: A Scoping Review," *Frontiers in Artificial Intelligence*, vol. 5, p. 15, Feb. 2022, doi: 10.3389/FRAI.2022.780405/BIBTEX.

[12] "Netflix Research." https://research.netflix.com/research-area/machine-learning (accessed Apr. 26, 2022).

[13] J. Wang, Y. Chen, V. W. Zheng, and M. Huang, "Deep Transfer Learning for Cross-domain Activity Recognition," *ACM International Conference Proceeding Series*, Jul. 2018, doi: 10.1145/3265689.3265705.

[14] G. Pinto, Z. Wang, A. Roy, T. Hong, and A. Capozzoli, "Transfer learning for smart buildings: A critical review of algorithms, applications, and future perspectives," *Advances in Applied Energy*, vol. 5, p. 100084, Feb. 2022, doi: 10.1016/J.ADAPEN.2022.100084.

[15] F. Zhuang *et al.*, "A Comprehensive Survey on Transfer Learning," 2020.

[16] M. Iman, K. Rasheed, and H. R. Arabnia, "A Review of Deep Transfer Learning and Recent Advancements," Jan. 2022, doi: 10.48550/arxiv.2201.09679.

[17] K. Dev, Z. Ashraf, P. K. Muhuri, and S. Kumar, "Deep autoencoder based domain adaptation for transfer learning," *Multimedia Tools and Applications*, pp. 1–27, Mar. 2022, doi: 10.1007/S11042-022-12226-2/TABLES/6.

[18] J. Zhang, W. Li, P. Ogunbona, and D. Xu, "Recent Advances in Transfer Learning for Cross-Dataset Visual Recognition," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, Feb. 2019, doi: 10.1145/3291124.

[19] Y. Fregier and J. B. Gouray, "Mind2Mind: Transfer Learning for GANs," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12829 LNCS, pp. 851–859, 2021, doi: 10.1007/978-3-030-80209-7_91.

[20] A. Shafahi *et al.*, "ADVERSARIALLY ROBUST TRANSFER LEARNING".

[21] Z. Zhu, K. Lin, and J. Zhou, "Transfer Learning in Deep Reinforcement Learning: A Survey," Sep. 2020, doi: 10.48550/arxiv.2009.07888.

[22] Y. Zhu, X. Wu, P. Li, Y. Zhang, and X. Hu, "Transfer learning with deep manifold regularized auto-encoders," *Neurocomputing*, vol. 369, pp. 145–154, Dec. 2019, doi: 10.1016/J.NEUCOM.2019.08.078.

[23] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010, doi: 10.1109/TKDE.2009.191.

[24] "Zero-shot Learning with Semantic Output Codes." https://proceedings.neurips.cc/paper/2009/hash/1543843a47 23ed2ab08e18053ae6dc5b-Abstract.html (accessed Apr. 27, 2022).

[25] B. Liu, "Lifelong machine learning: a paradigm for continuous learning," *Front. Comput. Sci*, vol. 11, no. 3, pp. 359–361, 2017, doi: 10.1007/s11704-016-6903-6.

[26] B. Koçer and A. Arslan, "Genetic transfer learning," *Expert Systems with Applications*, vol. 37, no. 10, pp. 6997–7002, Oct. 2010, doi: 10.1016/J.ESWA.2010.03.019.

[27] T. T. H. Dinh, T. H. Chu, and Q. U. Nguyen, "Transfer learning in Genetic Programming," *2015 IEEE Congress on Evolutionary Computation, CEC 2015 - Proceedings*, pp. 1145–1151, Sep. 2015, doi: 10.1109/CEC.2015.7257018.

[28] F. de L. Arcanjo, G. L. Pappa, P. v. Bicalho, W. Meira, and A. S. da Silva, "Semi-supervised genetic programming for classification," *Genetic and Evolutionary Computation Conference, GECCO'11*, pp. 1259–1266, 2011, doi: 10.1145/2001576.2001746.

[29] S. Mirjalili, H. Faris, and I. Aljarah, Eds., "Evolutionary Machine Learning Techniques," 2020, doi: 10.1007/978-981-32-9990-0.