Transfer Learning

MInDS @ Mines

Transfer learning is the sharing of knowledge between models. We can apply transfer learning to most, if not all, primary types of machine learning. Transfer learning is a large area of research and can produce great results by utilizing knowledge from state of the art models pre-trained on large datasets in conjunction with additional training for a target application.

Transfer Learning

Transfer learning is the sharing of knowledge between machine learning models. It can go by many other names such as: learning to learn, life-long learning, knowledge transfer, inductive transfer, multitask learning, knowledge consolidation, context-sensitive learning, knowledge-based inductive bias, metalearning, and incremental / cumulative learning. A formal definition of transfer learning can be provided as; given a source domain $\mathcal{D}_S$ and a learning task $\mathcal{T}_S$, a target domain $\mathcal{D}_T$ and a learning task $\mathcal{T}_T$, transfer learning aims to help improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$. We define a *domain* using a feature space and a probability distribution of the data. We define a *task* using a label space and a predictive function.[1]

[1] Sinno Jialin Pan, Qiang Yang, et al. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, pages 1345–1359, 2010

Based on this definition we get three main categories of transfer learning:

- Inductive Transfer Learning - where the target domain is known and the source is either known or unknown.
- Transductive Transfer Learning - where the source domain is known and the target domain is unknown.
- Unsupervised Transfer Learning - where neither the source domain nor target domain are known.

When we apply transfer learning another important aspect to think about is what exactly to transfer. This allows us to break down transfer learning approaches to a focus on transferring one of the following:

- Instance-based transfer learning - re-weighting samples in the source domain based on their relevance when applied to the target domain.
- Feature representation transfer learning - finding a feature representation that reduces the difference between the source and target domains.
- Parameter transfer learning - determining shared parameters between the domains for the trained models.
- Relational knowledge transfer learning - mapping relational knowledge between source and target domains.

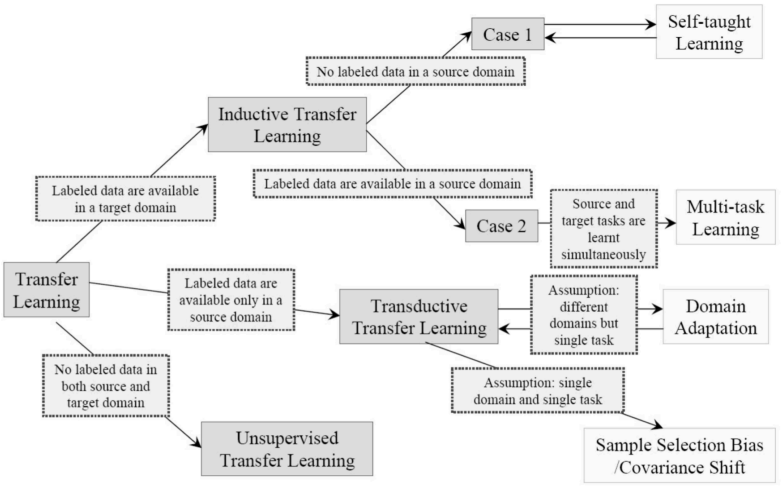

Figure 1: An illustration of the categories of transfer learning based on domain knowledge.

These approaches don't apply to all cases of transfer learning. We compare the three main types of transfer learning and their ability to apply the different approaches in the following table:

What to transfer / Category	Inductive	Transductive	Unsupervised
Instance	✓	✓	
Feature Representation	✓	✓	✓
Parameters	✓		
Relational Knowledge	✓		

Table 1: Applicable approaches for the categories within transfer learning.

When and why to use transfer learning

We've looked at what transfer learning is and the different approaches we can apply. Now we take a look at when and why to consider transfer learning for our applications. There are several reasons to consider transfer learning. First, we could be interested in augmenting our data with a similar domain. A good example of this is in natural language processing where we may want to use one language's models as a starting point to a different language's models. The distribution of data may be similar yet different but it gives us a much better start at the model since as humans we are likely to discuss the same topics regardless of language.

Second, we might be solving a machine learning problem in a niche area where the amount of available data is minimal. For this approach, we would want to utilize any knowledge that we can gain from larger datasets in a similar domain and start there. This simulates the scenario in which our model goes out into the world and has acquired some previous knowledge

that they can bring to our specific application.

Finally, we might want to ensure that our model does not overfit to the data we have provided. By utilizing transfer learning, the model would balance between its source domain and the target domain it is being trained on again. This makes it more likely that the model will learn a more general understanding of the data instead of focusing on the details within the data and overfitting.

Effectiveness of transfer learning

Transfer learning may seem intuitive but it's not really clear how effective it may be to make it worthwhile to use and spend the time on. Some example results on a particular dataset, the 20 news groups text dataset, from 2010 show the performance difference between using a simple baseline / traditional machine learning approach versus a transfer learning approach.

Source vs Target	SVM Accuracy (Baseline)	TrAdaBoost Accuracy (Transfer)
rec vs talk	87.3%	92.0%
rec vs sci	83.6%	90.3%
sci vs talk	82.3%	87.5%

Table 2: Example results comparing baseline methods to transfer learning methods on the 20 Newsgroups dataset for the recreation, science and talk labels.

Issues with transfer learning

One main issue to be aware of with transfer learning is what is called *negative transfer learning*. Negative transfer learning is when the knowledge being transferred can negatively impact the model focused on the target domain and task.

Application-specific transfer learning

Now that you hopefully understand the general premise behind transfer learning, we can take a closer look at how we can apply it to specific applications. With a variety of applications, the community has been able to collect and preserve large datasets that are representative of a particular application type such as object detection. Because of these datasets, we can produce significantly better results on a variety of related tasks by starting with models that were pre-trained on these datasets.

One key advantage of transfer learning is the ease at which we can use state of the art models pre-trained on large datasets to our target application.

The most commonly used approach for transfer learning in practice is the feature representation transfer learning since it applies to all types of applications. We caa use any pre-trained feature learning model and apply its learnings to a new dataset, assuming they follow a similar input feature space. Another commonly used approach is parameter transfer learning. This

approach is commonly used with deep learning since we can use a network with an already successful architecture and initialize our weights at a point that has already detected useful patterns in data. We can also look at an approach that combines feature transfer learning and parameter transfer learning; copying the portion of a model, usually a neural network, that learns a feature representation with its pre-trained parameters and then including other target domain specific models following that in our pipeline.

Computer Vision

With regards to Computer Vision (CV), the ImageNet dataset[2] is one of the largest openly available datasets. It is also considered a benchmark dataset for new machine learning models to compare their performance. Some effective models for object detection and image representation in CV include Xception[3], VGG-16 and VGG-19[4], and ResNet50[5]. Since these models are Convolutional Neural Networks, we can break apart the portion that focuses on feature representation (the convolutional layers) and plug in our own neural network following that. This approach allows us to use a powerful architecture with already detected patterns since it was already trained on a large dataset.

Natural Language Processing

Language is inherently a supervised learning problem in that we must learn the language's meanings before we can start to have a conversation using the same words. Each word has some value and reference point in the real world. We can simulate this supervision of teaching the model a language by using transfer learning with a pre-trained model on a large dataset. Some commonly used datasets in natural language processing (NLP) are the Common Crawl dataset[6], Google Books, and Wikipedia. Some effective models for NLP include GloVe[7] which provide word embeddings to be used for text, and other models which can provide feature representations that are more context-aware, such as ELMo[8], GLoMo[9] and the OpenAI Transformer[10].

[2] The ImageNet dataset consists of over 14 million images and 20 thousand synsets or synonym sets. For more information, visit http://image-net.org

[3] F Chollet. Xception: Deep learning with depthwise separable convolutions. arxiv: 161002357. 2016

[4] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014

[5] K He, X Zhang, S Ren, and J Sun. Deep residual learning for image recognition. corr, vol. abs/1512.03385, 2015

[6] The Common Crawl dataset is collected text across the internet amounting to over 2.5 billion web pages. For more information, visit http://commoncrawl.org/

[7] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014

[8] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proc. of NAACL*, 2018

[9] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018

[10] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018