Transfer Learning & Domain Adaptation

Presented by: CHEA Navy
03/07/2020
Artificial Intelligence Laboratory Dongguk University

Content

- Overview of Transfer Learning
- Notations and Definitions
- Transfer Learning Techniques
- Transfer Learning for Deep Learning
- Domain Adaptation
- Conclusion

Overview

A brief history of Transfer Learning

- Know how to ride a motorbike, learn how to ride a car
- Know how to play classic piano, learn how to play jazz piano

- know math and statistics, learn machine learning

Overview

Transfer Learning

A brief history of Transfer Learning

Traditional Machine Learning

Overview

A brief history of Transfer Learning

Overview

A brief history of Transfer Learning

- Andrew Ng, NIPS 2016 “Nuts and Bolts of

building AI
applications using Deep Learning”

“After supervised learning, Transfer Learning will be the next driver of ML commercial success”

○ “Deep Learning” ~ Ian Goodfellow et al

“Situation where what has been learned in one setting is exploited to improve generalization in another setting.”

6

Notations and Definitions

Suppose we have: a domain $D = \{X, \mathbf{P(X)}\}$,

where

- *X:* Feature Space
- **P(X)**: Marginal probability distribution
- **X = {**x_1, ..., x_n} $\in$ *X* (Particular learning samples)

Notations and Definitions

Suppose we have: a task *T = {Y, f(.)}* , where

- *Y :* Label Space

○ $f(.)$: Objective predictive function

○ Which can be learned from training data $\{x_i , y_i\}$,

$x_i \in \boldsymbol{X}$ **&** $y_i \in Y$ ○ $f(.)$: predicts the $f(x)$

○ $f(x) = \mathbf{P}(y|x)$

We consider with Source domain **(**D_S) & Target

domain **(**D_T) and their learning tasks T_S & T_T .

Notations and Definitions

❑ Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T, transfer learning aims to help improve the learning of the

target predictive function $f_T(.)$ in D_T using the knowledge in D_S and T_S, where $D_S \neq D_T$, or $T_S \neq T_T$

Transfer Learning Techniques

Main research issues in Transfer Learning:

○ **"What to transfer"**, which part of knowledge to transfer o **“When to transfer”**, which situation that transferring should be done

o **“How to transfer”**, go forward for transferring the knowledge across

the domain.

10

Transfer Learning Techniques

11

Transfer Learning Techniques

Transfer Learning Setting Related Area **Source Domain**

Labels

Target Domain

Labels **Tasks**

Inductive Transfer Learning

Multi-task Learning Available Available Classification

Regression,

Self-taught Learning Unavailable Available

Regression, Classification

Transductive Transfer Learning

Domain Adaptation, Sample Selection Bias, Co-variate Shift

Available Unavailable

Classification Regression,

Unsupervised Transfer

Learning Unavailable Unavailable

Clustering, Dimensionality Reduction

Transfer Learning Techniques

Inductive Transfer Learning

Given a source domain D_S and learning task T_S, a target domain D_T and learning task T_T, transfer learning aims to help improve the learning of the target predictive function $f_T(.)$ in D_T using the knowledge in D_S and T_S, where $T_S \neq T_T$

13

Transfer Learning Techniques

Transductive Transfer Learning

○ Given a source domain D_S and learning task T_S, a target

domain D_T and learning task T_T**,** transfer learning aims to help improve the learning of the target predictive function $f_T(.)$ in D_T using the knowledge in D_S and T_S , where $D_S \neq D_T$, $T_S = T_T$

○ Note that some unlabeled target domain data must be available at training time

Transfer Learning Techniques

Unsupervised Transfer Learning

○ Given a source domain D_S and learning task T_S**,** a target

domain D_T and learning task T_T, transfer learning aims to help improve the learning of the target predictive function $f_T(.)$ in D_T using the knowledge in D_S and T_S, where $T_S \neq T_T$, and Y_S and Y_T are not observable.

○ Note that there's no labeled data in both source and target domain.

Transfer Learning Techniques

Different Approaches used in different settings

- Instance Transfer
- Feature-representation Transfer
- Parameter Transfer
- Rational-knowledge Transfer

16

Transfer Learning for Deep Learning

Myth: you cannot do deep learning unless you have a million of

labeled examples for your problem.

Reality

○ You can learn useful representations from unlabeled data

○ You can train on a nearby surrogate objective for which it is easy

to generate labels

○ You can transfer learned representations from a related task.

Transfer Learning for Deep Learning

Instead of training a deep network from scratch for your task:

○ Take a network trained on a different domain for a different

source task

○ Adapt it for your domain and your target task

We will talk about how to do this.

Variations:

- Same domain, different task
- Different domain, same task

Transfer Learning for Deep Learning *PASCAL VOC 2007*

○ 20 classes, ~10K images, 50% train, 50% test ○ Deep networks can have many parameters (e.g. 60M in Alexnet) ○ Direct training (from scratch) using only 5K training images can be

problematic. Model overfits. ○ How can we use deep networks in this setting?

Transfer Learning for Deep Learning *“Off-the-shelf”*

use outputs of one or more layers of a network trained on a different task as generic feature detectors. Train a new shallow model on these features.

Transfer Learning for Deep Learning *“Off-the-shelf”*

Surpassed or on par with state-of-the-art in several tasks in 2014

Image classification:

- PASCAL VOC 2007

- Oxford flowers
- CUB Bird dataset
- MIT

indoors Image retrieval:

- Paris 6k
- Holidays
- UKBench

Razavian et al, CNN Features off-the-shelf: an Astounding Baseline for Recognition, CVPRW 2014: https://arxiv.org/pdf/1403.6382.pdf

Domain Adaptation

Consider a classification task where X is the input space & Y is the set of labels. Given two sets of samples drawn from the source & target domains.

$D_S = \{ x_i, y_i \}^n{}_{i=1} \sim P(X_S)$, $D_T = \{ x_i, y_i \}^n{}_{i=n+1} \sim P(X_T)$

(Target space labeled data may not be present in unsupervised case)

Domain Adaptation

The goal of the learning algorithm is to build a

classifier $\eta: X \rightarrow Y$ with a low target risk

$R_{DT}(\eta) = {}_{x,y}\, Pr\ {}_{\sim DT}(\eta\ x \neq y)$ 23

Domain Adaptation *Domain Biases*

- Datasets are samples of the world

- In many cases, there is a shift or bias in the distributions of the

source and target data representations

i.e. when recognizing people, the target domain typically contains one

person centered with minimal background clutter, whereas the source dataset contains many people with more clutter. Thus, the neurons that

capture the features of other people and clutter are useless.

Domain Adaptation *Domain Shift*

○ The size of the shift is often measured by the distance between source &

target subspaces

○ A typical approach is to learn a feature space transformation to align the

source & target representation (reduce domain divergence)

Domain Adaptation

Fine-tuning: supervised domain adaptation

Train deep net on “nearby” task for which it is easy to get labels using standard backprop

Real loss

- E.g. ImageNet classification
- Pseudo classes from augmented data
- Slow feature learning, ego-motion

Cut off top layer(s) of network and replace

with supervised objective

for target domain.

Fine-tune network using backprop with

labels for target domain until validation loss

starts to increase.

Real data Real labels

My_fc2 + softmax

fc1

conv3

conv2

conv1

Domain Adaptation *Freeze or Fine-tune?*

LR > 0

loss

Bottom n layers can frozen or fine tuned

d fc2 + softmax e

- Frozen: not updated during backprop

nute

- Fine-tuned: updated during backprop

fc1

ni F conv3 **Which to do depends on target task:**

n

e conv2 z ○ Freeze: target task labels are scarce, and

or F we want to avoid overfitting

conv1 ○ Fine-tune: target task labels are more plentiful

LR = 0

data labels

Domain Adaptation *How transferable are features?*

Lower layers: more general features. Transfer very well to other tasks.

Higher layers: more task specific Fine-tuning improves generalization when sufficient examples are available.

Transfer learning and fine tuning often lead to better performance than training from scratch on the dataset.

Yosinki et al. How transferable are features in deep neural networks. NIPS 2014. https://arxiv.org/abs/1411.1792

Domain Adaptation *Unsupervised Domain Adaptation*

It is possible to do domain adaptation with labeled data point in target set.

Y Ganin and V Lempitsky, Unsupervised Domain Adaptation by Backpropagation, ICML 2015 https://arxiv.org/abs/1409.7495

Conclusion

Transfer Learning is the ability to apply the knowledge learned in previous Tasks to novel tasks.

▪ Possible to train very large models on small data by using Transfer Learning and Domain Adaptation

▪ Off the shelf features work very well in various domains and tasks

▪ Lower layers of network contain very generic features, higher layers more task specific features

▪ Example for supervised domain adaptation via fine tuning almost always improves performance

▪ Possible to do unsupervised domain adaptation by matching feature distributions

References

Contents:

○ A Survey on Transfer Learning, Sinno Jialin Pan and Qiang Yang *Fellow, IEEE*

○ A survey of transfer learning, Karl Weiss*, Taghi M. Khoshgoftaar and DingDing Wang

○ “Deep Learning”, Ian Goodfellow at el.

○ Andrew Ng, NIPS 2016 “Nuts and Bolts of building AI applications using Deep Learning”

○ DLCV D2L5 Transfer Learning and Domain Adaptation Tutorial

○ A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep

Learning, Dipanjan (DJ) Sarkar

Images:

○
https://www.einfochips.com/blog/how-does-transfer-learning-speeds-up-deep-learning-projects/

Thank You!