Deep Transfer Learning For Image Classification

Nguyen Quang Uy

Le Quy Don Technical University
AI Research, VNG Company

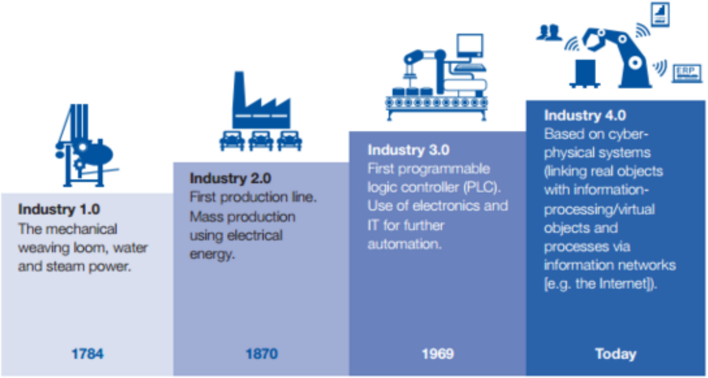
March 28, 2019

Outlines

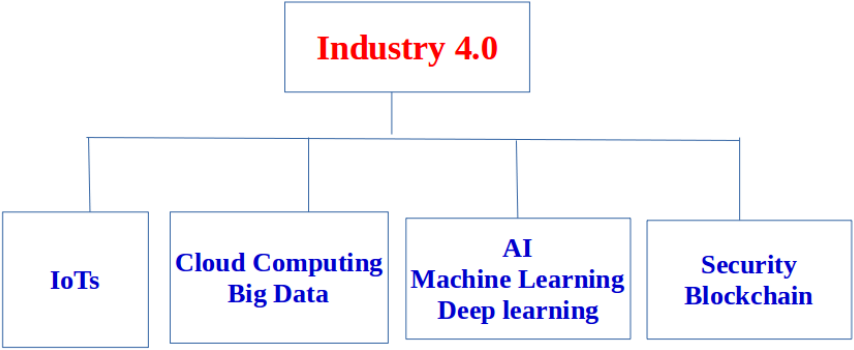
1. **The Industry 4.0 and Deep learning**
2. What is transfer learning
3. Transfer learning approaches
4. Transfer learning in practice

The Industry Resolutions

- Human beings are currently in the progress of the Industry 4.0

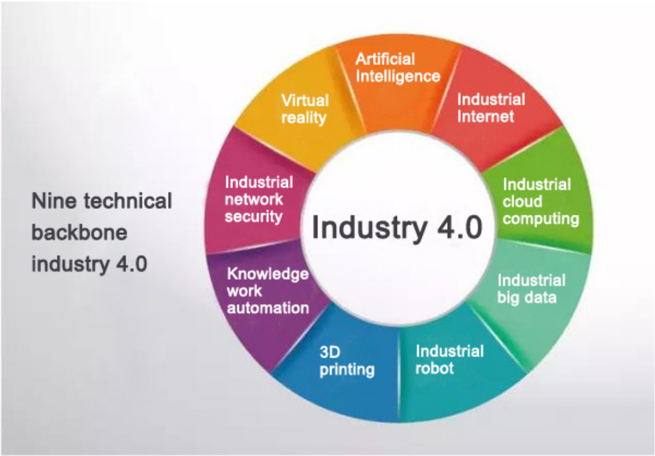


Four most important components of the Industry 4.0



AI and Industry 4.0

- Artificial Intelligence is considered as the Brains behind Industry 4.0



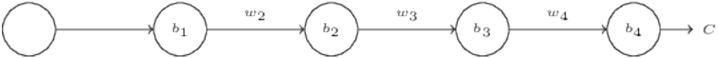
What is deep learning?

- (Wiki), **Deep learning** is a class of machine learning algorithms that:
  - Use a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each successive layer uses the output from the previous layer as input.
  - Learn in supervised (e.g., classification) and/or unsupervised (e.g., pattern analysis) manners.
  - Learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.

Training deep neural networks

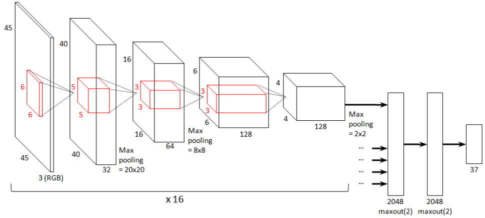
- Training fully connected deep neural network is very challenging:
  - **Vanishing gradients** problem makes lower layers very hard to train.
  - Training would be **extremely slow**.
  - A model with **millions of parameters** would severely risk **overfitting** the training set.

$$\frac{\partial C}{\partial b_1} = \sigma'(z_1) \times w_2 \times \sigma'(z_2) \times w_3 \times \sigma'(z_3) \times w_4 \times \sigma'(z_4) \times \frac{\partial C}{\partial a_4}$$



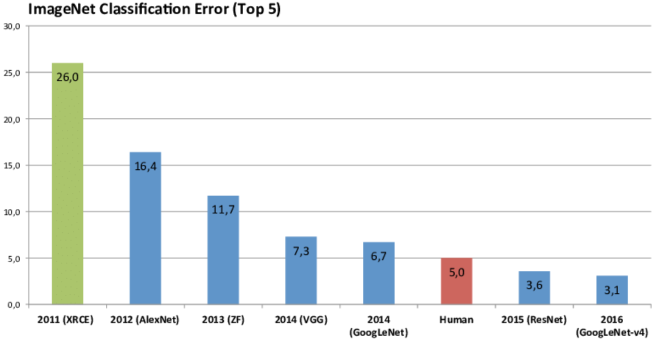
Deep learning

- Deep learning is training the neural network with many layers and with some important modifications.
  - **Network structure**: To reduce parameters and avoid overfitting problem.
  - **Training algorithms**: to speed up the training process
  - **Activation function**: To avoid the gradient vanishing problem.



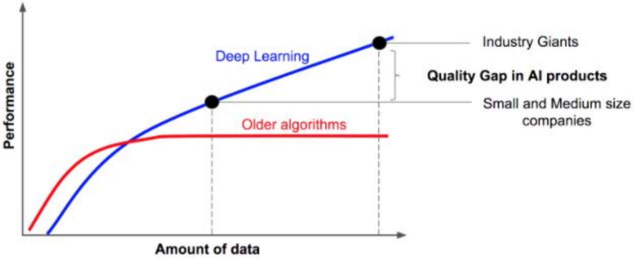
Deep learning gradually beat human in image classification

- Imagenet contains more than 1 million images of 1000 categories.



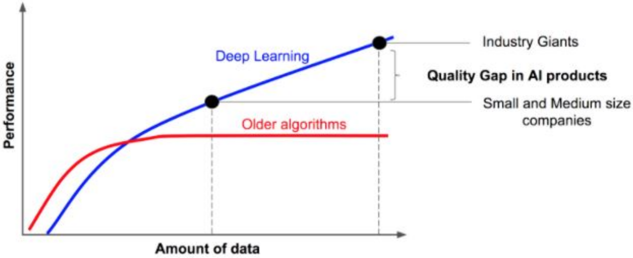
The performance of deep learning vs data size

- A large quantity of input data is required to train a good deep learning model.



Transfer learning assists deep learning in practice

- Transfer learning has been used to relaxes the massive data requirement of deep neural networks.

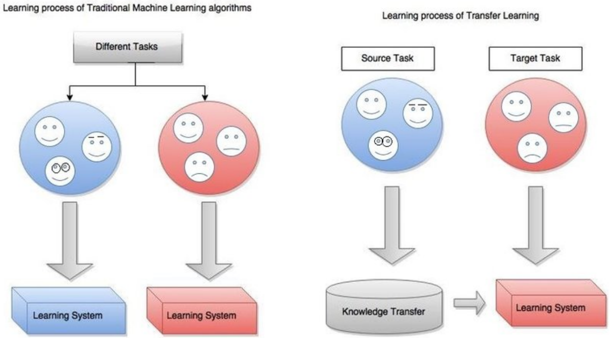


Outlines

1. The Industry 4.0 and Deep learning
2. **What is transfer learning**
3. Transfer learning approaches
4. Transfer learning in practice

Traditional machine learning

- In classic machine learning, if we intend to train a model for a task A, we assume that we have enough data for that task.
- When given data for task B, we require data of this task to train a new model B and we hope it perform well on this data.

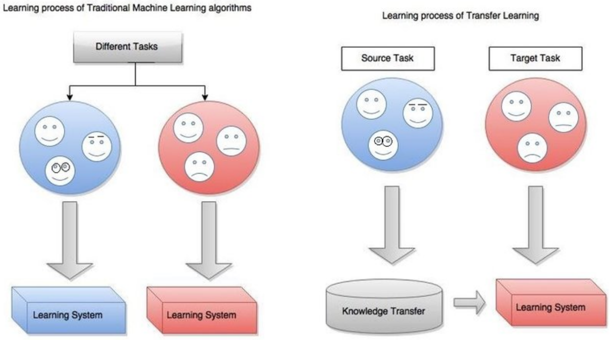


Traditional machine learning

- The traditional learning breaks down when we do not have sufficient data: A model to detect pedestrians on day-time images could not apply to night-time images.
- This happens for a variety of reasons, which we can term as the model's bias towards training data and domain.

Transfer learning

- Transfer learning should enable us to utilize knowledge from previously learned tasks and apply them to newer, related ones.



Transfer learning

- Transfer learning is more similar to human learning than isolated learning.
- Humans have an inherent ability to transfer knowledge across tasks.

The definition of transfer learning

- Transfer learning involves the concepts of a **domain** and a **task**.
- A domain **D** consists of a feature space **F** and a marginal distribution **P(X)** over the feature space, where $\mathbf{X{=}x1,\cdots,xn}\in\mathbf{F}$.
- For document classification with a bag-of-words representation:
  - **F** is the space of all document representations,
  - **xi** is the i-th term vector corresponding to some document and
  - **X** is the sample of documents used for training.

The definition of transfer learning

- Given a **domain**, $\mathbf{D=\{F,P(X)\}}$, a **task T** consists of a label space $\mathbf{Y}$ and a conditional distribution $\mathbf{P}(\mathbf{Y} \mid \mathbf{X})$.
- Typically, $\mathbf{P}(\mathbf{Y} \mid \mathbf{X})$ is learned from the training data consisting of pairs $\mathbf{xi} \in \mathbf{X}$ and $\mathbf{yi} \in \mathbf{Y}$.
- In our document classification example, **Y** is the set of all labels, i.e. True, False and **yi** is either True or False.

The definition of transfer learning

- Given a source domain $\mathbf{D_S}$, a source task $\mathbf{T_S}$, a target domain $\mathbf{D_T}$, and a target task $\mathbf{T_T}$,
- Transfer learning aims to learn the target conditional distribution $\mathbf{P}(\mathbf{Y_T}|\mathbf{X_T})$ in $\mathbf{D_T}$ with the information gained from $\mathbf{D_S}$ and $\mathbf{T_S}$ where $\mathbf{D_S} \neq \mathbf{D_T}$ or $\mathbf{T_S} \neq \mathbf{T_T}$.
- In most cases, the number of target examples is exponentially smaller than the number of source examples.

Four scenarios of transfer learning

- If two domains are different, they may have different feature space or different marginal distribution.
  - The documents are written in two different languages.
  - The documents discuss different topics.
- If two tasks are different, they may have different labeled space or different conditional distribution.
  - The documents are assigned to different classes.
  - The documents in the source and the target are imbalanced.

Outlines

Deep transfer learning

- There are two main approaches to applying transfer learning in deep learning:
  - Using a pre-trained model
  - Developing a new model

Using a pre-trained model

- 1. Select a source model: A pre-trained source model is chosen from available models.
- 2. Use pre-trained model as a fixed feature extractor.
- 3. Retrain model: the model may need to be adapted or refined on the input-output pair data available for the task of interest.

Retrain model

- It is possible to fine-tune all the layers of the ConvNet, or it's possible to keep some of the earlier layers fixed and only fine-tune some higher-level layers.
- The earlier features often contain more generic features (e.g. edge detectors or color blob detectors) that should be useful to many tasks.
- The later layers becomes progressively more specific to the details of the classes contained in the original dataset.

Develop Model Approach

- 1. Select source task: You must select a related problem with an abundance of data.
- 2. Develop source model: Next, you must develop a model for this first task. The model must be better than a naive model to ensure that some feature learning has been performed.
- 3. Transfer the trained model from the source to the target.

Outlines

1. What is transfer learning
2. Transfer learning approaches
3. **Transfer learning in practice**

Deep transfer learning in practice

- How do you decide what type of transfer learning you should perform on a new dataset?
- This is a function of several factors, but the two most important ones are:
  - The size of the new dataset (small or big)
  - Its similarity to the original dataset (e.g. ImageNet-like in terms of the content of images and the classes, or very different, such as microscope images).

Transfer learning in practice

- 1. New dataset is small and similar to original dataset:
  - Since the data is small, it is not a good idea to fine-tune the ConvNet due to overfitting concerns.
  - Since the data is similar to the original data, we expect higher-level features in the ConvNet to be relevant to this dataset as well.
  - Hence, the best idea might be to train a linear classifier on the CNN codes.

Transfer learning in practice

- 2. New dataset is large and similar to the original dataset:
  - Since we have more data, we can have more confidence that we won't overfit if we were to try to fine-tune through the full network.
- 3. New dataset is small but very different from the original dataset.
  - Since the data is small, it is likely best to only train a linear classifier.
  - Since the dataset is very different, it might not be best to train the classifier form the top of the network, which contains more dataset-specific features.
  - Instead, it might work better to train the SVM classifier from activations somewhere earlier in the network.

Transfer learning in practice

- 4. New dataset is large and very different from the original dataset:
  - Since the dataset is very large, we may expect that we can afford to train a ConvNet from scratch.
  - However, in practice it is very often still beneficial to initialize with weights from a pretrained model.
  - In this case, we would have enough data and confidence to fine-tune through the entire network.

Practical advice

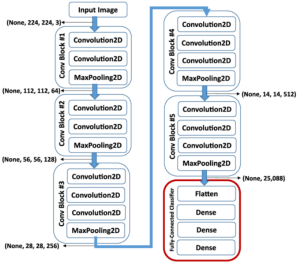
- There are a few additional things to keep in mind when performing Transfer Learning:
- Constraints from pretrained models. Note that if you wish to use a pretrained network, you may be slightly constrained in terms of the architecture you can use for your new dataset.

Practical advice

- Learning rates. It's common to use a smaller learning rate for ConvNet weights that are being fine-tuned, in comparison to the (randomly-initialized) weights for the new linear classifier that computes the class scores of your new dataset.
- This is because we expect that the ConvNet weights are relatively good, so we don't wish to distort them too quickly and too much (especially while the new Linear Classifier above them is being trained from random initialization).

Transfer learning in Keras

- We will use the VGG16 architecture, pre-trained on the ImageNet dataset .



Thank you for your attention!