

Deep Learning for Image Analysis - Weak supervision and learning from simulations

Thomas Walter, PhD

Centre for Computational Biology (CBIO)
MINES Paris-Tech, PSL Research University
Institut Curie, PSL Research University
INSERM U900

Overview

- 1 Motivation
- 2 Strategies
- 3 Weak supervision
- 4 Learning from simulated data
- 5 Conclusion
- 6 References

Overview

1 Motivation

2 Strategies

3 Weak supervision

4 Learning from simulated data

5 Conclusion

6 References

Deep Learning: there is just one catch ...

- Deep Learning is today the most powerful method for image classification, segmentation and object detection.
- Deep Learning can achieve or even go beyond human performance for these tasks.
- There is a snag: deep learning relies on massive annotation.
- Example: ImageNet contains 1.4 million annotated images [Russakovsky et al., 2015].

Deep Learning: there is just one catch ...

- Manual image annotation is annoying.
- Manual image annotation is expensive.
- We need to address the need for massive image data sets, either by avoiding massive annotation or by making annotation cheap.

Two main strategies to overcome massive image annotation

- 1 Use cheap annotations.
- 2 Use different annotations.

Please note:

- We always need to consider what exactly we aim at predicting (e.g. the value of a pixel or the class of an image)
- For instance, in segmentation, annotations are expensive at the image level, but not at the pixel level: a single stroke annotates hundreds of pixels.

Image segmentation: cheap annotations?

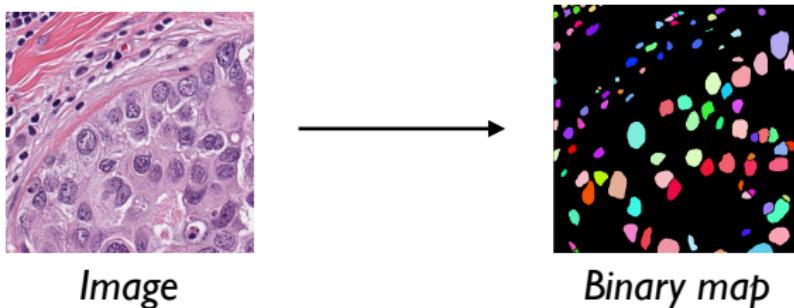


Figure: Segmentation is tedious, but provides rapidly many samples.

- Providing pixel-level annotations is extremely tedious.
- But as we are classifying pixels, we can rapidly provide hundreds of thousands of samples.
- In practice however, we need to **cover the variability** to be expected at prediction time.
- For this reason, we often need many images (depending on the variability of the images).

Massive image annotation by crowd sourcing



Figure: Gamification for the annotation of protein localization patterns: the actual annotation task is "disguised" as a computer game. Image taken from [Sullivan et al., 2018]

- Crowd sourcing: generate massive annotated data sets by recruiting more people to do the annotations.
- This requires the implication of untrained experts (citizen science).
- Several strategies to reach many people, e.g. gamification.

Massive image annotation by leveraging routinely acquired data

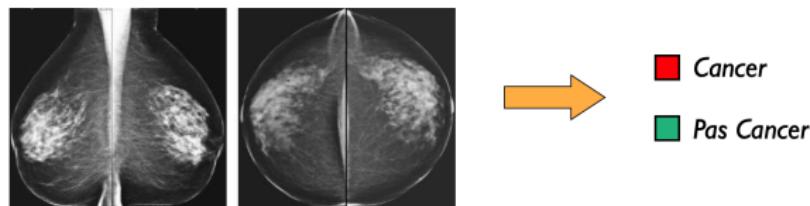


Figure: Radiology: mammographies are acquired routinely by physicians.

- In many fields, image data is routinely acquired (e.g. medical examinations).
- Origins of the image labels:
 - Routine annotation by a medical doctor.
 - Future evolution of the disease.
- Problem: we are limited in the tasks, i.e. in the variables that we can predict.

Massive image annotation by experimental setup

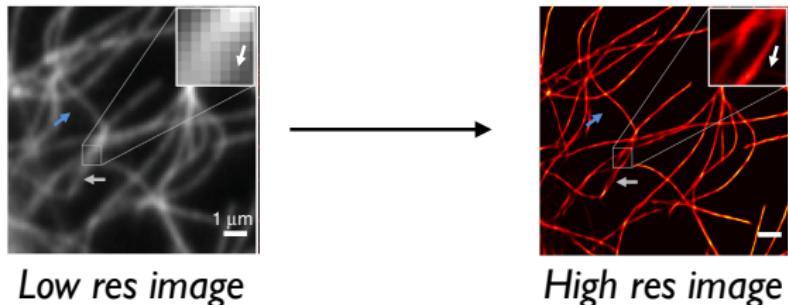


Figure: A high resolution image can be predicted from a low resolution image. Image adapted from [Ouyang et al., 2018]

- In image restoration, we aim at predicting a high quality image from a lower quality image.
- Here, the ground-truth is entirely generated by the experimental setup (pairs of images are acquired).

Massive image annotation by experimental setup

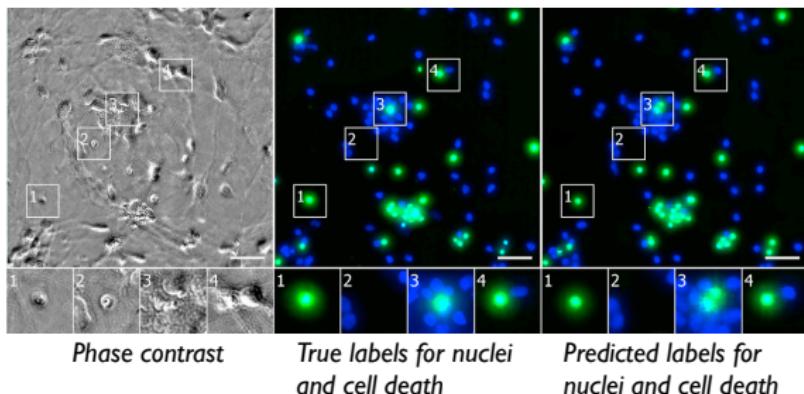


Figure: An image taken with a different modality can be predicted.

Here, we show an example for a fluorescent microscopy image predicted from a phase contrast microscopy image. Image adapted from [Christiansen et al., 2018]

- We can also predict images taken with different modalities.
- This is interesting if one technique is more expensive or more invasive.
- Similarly, we can also use different experimental techniques to generate ground truth experimentally [J. Boyd et al., 2020].

Overcoming massive image annotation: simulation

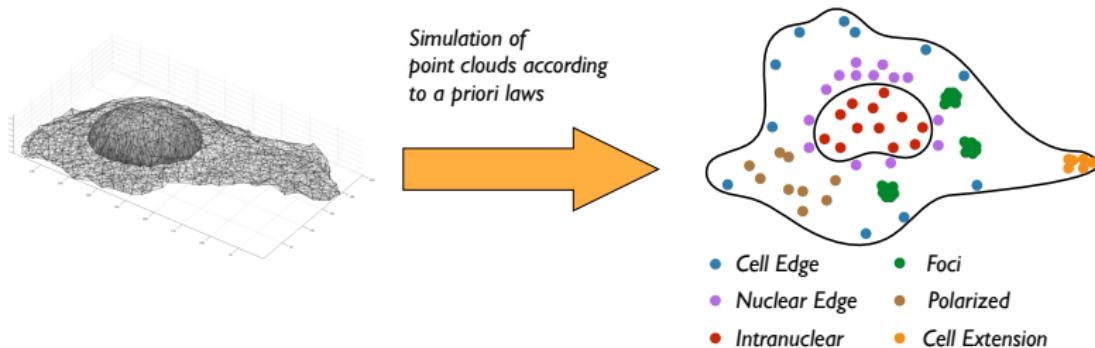


Figure: Simulated RNA localization in cells

- Simulation of large amounts of data with known ground truth.
- Train networks on simulated data.
- Problem: the data distributions between simulated and real data usually differ.

Overcoming massive image annotation: transfer learning

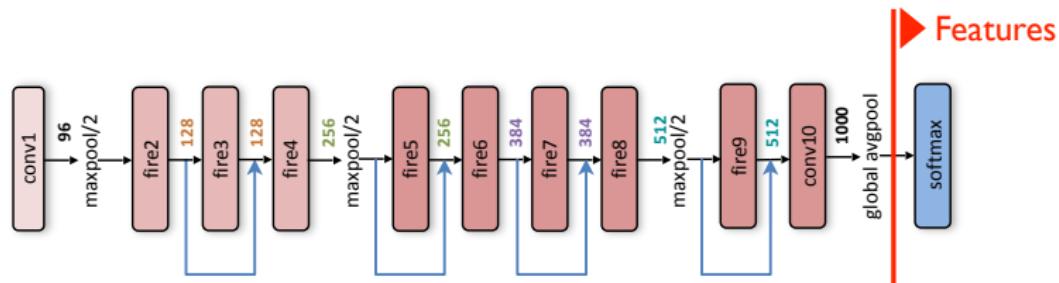


Figure: Image representations for transfer learning. Image adapted from [Iandola et al., 2017]

- Train a network on large annotated image bases.
- Use the learned representations to solve small scale problems (with or without fine-tuning) with few annotations.

Subject of today

In this lecture, we will learn about two strategies to address the problem of massive datasets required for training of deep neural networks:

- 1 Learning in a weakly supervised setting (typically coarse annotations)
- 2 Learning from simulations

Overview

- 1 Motivation
- 2 Strategies
- 3 Weak supervision
- 4 Learning from simulated data
- 5 Conclusion
- 6 References

What is weak supervision?

- Weak supervision refers to a situation, where the ground truth data we build our model on is in some sense imperfect [Zhou, 2018].
- Three types of weak supervision:
 - 1 **Incomplete supervision:** only a subset of training data are labeled (semisupervised setting).
 - 2 **Inexact supervision:** training data are labeled but not as exact as required by the task we would like to perform
 - 3 **Inaccurate supervision:** training data are labeled, but may contain mistakes () .
- Here, we treat the case of inexact supervision.

The Multiple Instance Learning framework (MIL)

- MIL deals with problems with incomplete knowledge of labels in training sets.
- MIL assumes that instead of disposing of individual labels y_i for each sample x_i , we only have labels for subsets of samples, called "bags":

$$\begin{aligned} T &= \{(B_i, y_i)\} \\ B_i &= \{x_j\} \end{aligned} \tag{1}$$

- A bag is positively labeled if at least one instance in it is positive, and is negatively labeled if all instances in it are negative.
- Examples:
 - An image is labeled as "contains humans", if at least one region contains a human.
 - A tissue slide is labeled cancerous, if at least one region is cancerous.

A deep learning approach to MIL

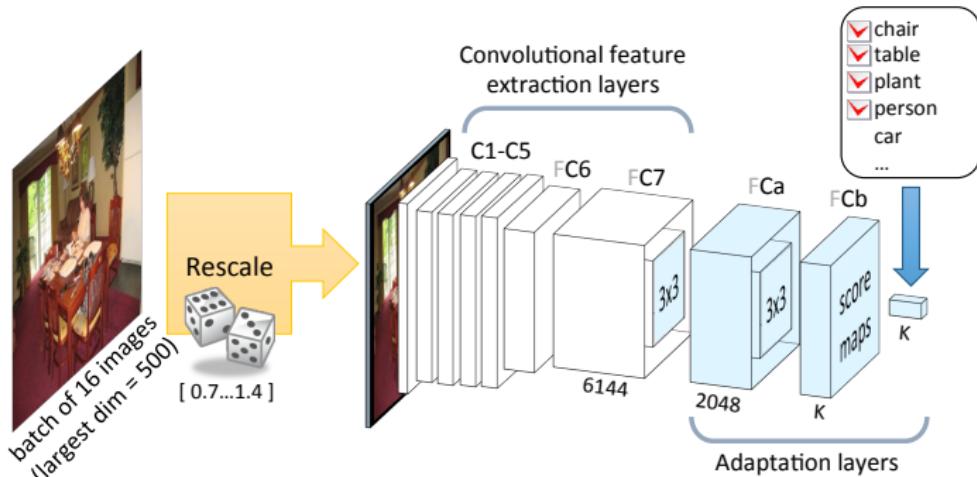


Figure: CNN architecture for Multiple Instance Learning. Image taken from [Oquab et al., 2015]

First, a standard CNN is applied as feature extractor (without fully connected layers). This maps an image X to a $n \times m \times l$ layer (l feature maps of spatial dimensions $n \times m$).

A deep learning approach to MIL

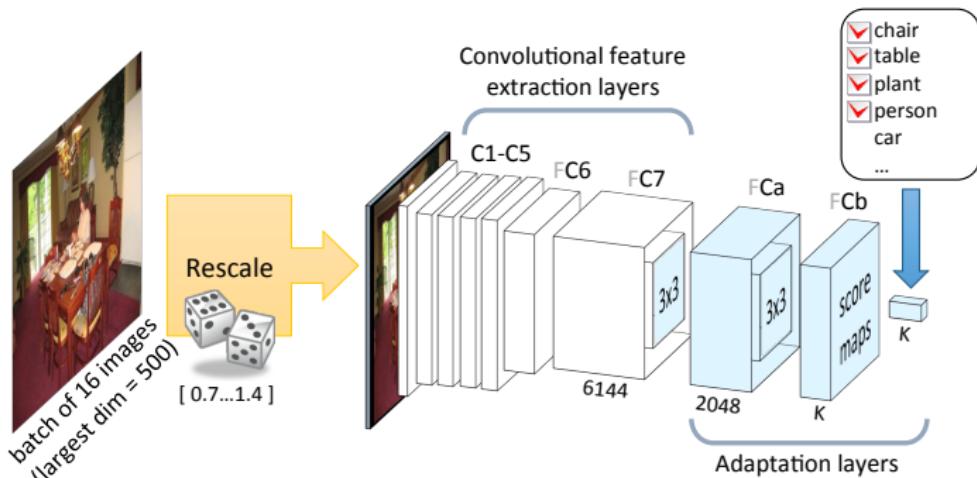


Figure: CNN architecture for Multiple Instance Learning. Image taken from [Oquab et al., 2015]

This layer is then mapped to a $n \times m \times K$ layer by 1×1 convolutions, where K is the number of output classes. Of note, a 1×1 convolutional layer is equivalent to a fully connected layer, if its input is a vector (spatial dimension 1). We can understand the 1×1 conv layer as parallel fully connected layers.

A deep learning approach to MIL

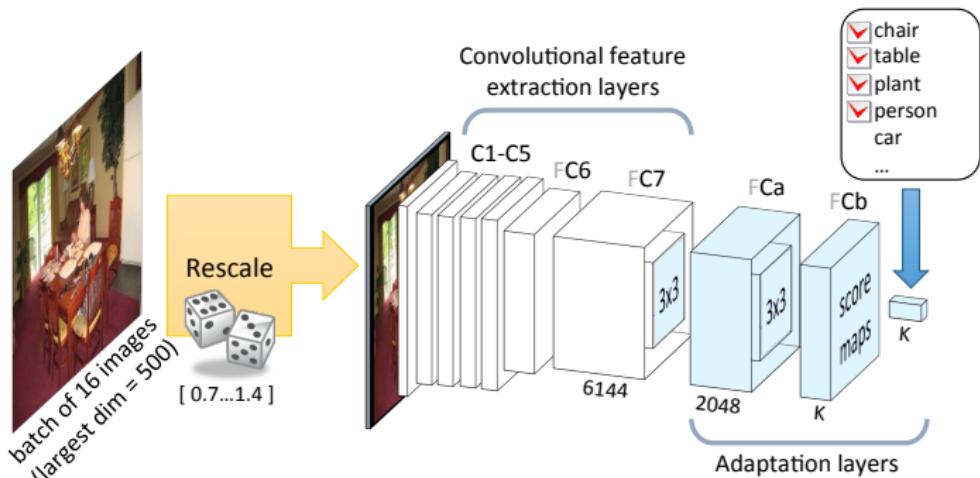


Figure: CNN architecture for Multiple Instance Learning. Image taken from [Oquab et al., 2015]

Each "pixel" in feature map FCb is thus the score vector for an input region. FCb therefore corresponds to a bag of image regions. We therefore maximize over the spatial dimensions:

$$s_k = \max_{i,j} f_{i,j}^k \quad (2)$$

A deep learning approach to MIL



Figure: Example for MIL by deep learning. Image adapted from [Oquab et al., 2015]

- Maximization over the spatial dimensions corresponds exactly to the MIL-paradigm: for the decision on the image label, it is sufficient to have one single positive region.
- This is a typical situation, when we want to detect objects in crowded images.

Discussion

- Improvement of image classification, as only the relevant region is taken into consideration.
- This is particularly useful, if the size of the relevant region is comparatively small.
- Objects can also be detected and this without expensive annotation (bounding boxes or pixel-wise annotation).
- In addition, the detection of different classes is eased by the setup.
- Limitations:
 - Scale issue: the desired object might not fit into the considered regions.
 - Object extensions cannot be faithfully predicted.
 - The context is not taken into consideration.

Improvements on the WSL strategy

- Instead of taking the maximum score over all image regions [Durand et al., 2016]:
 - Select the s_{top} highest scoring regions
 - Select the s_{low} lowest scoring regions
- Intuition: taking $s_{top} > 1$ regions makes the algorithm more robust, taking s_{low} lowest scoring regions provides negative evidence for the class.

Application example: histopathology

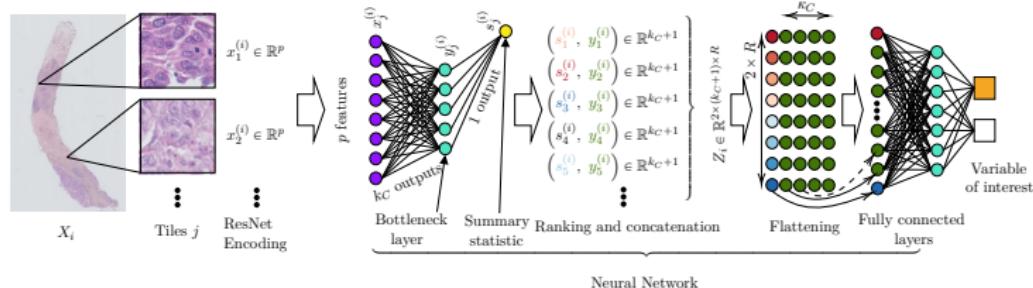


Figure: Example for WSL for tumor detection. Image provided by Peter Naylor.

- Histopathology images are extremely big.
- Sometimes, what we want to find can be small.
- There is slide-level annotation, but detailed annotation is scarce.

Application example: metastasis detection

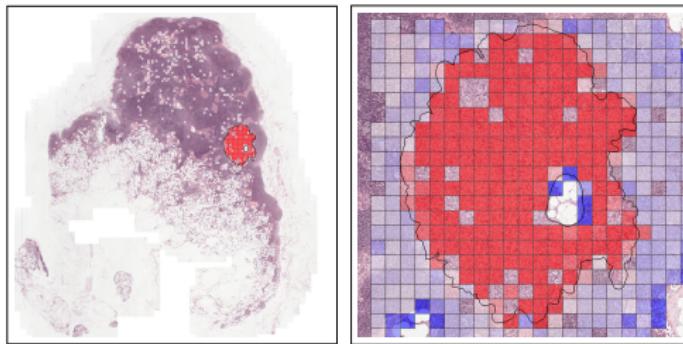


Figure: Example for WSL for tumor detection. Image adapted from [Courtiol et al., 2017]

Here, [Courtiol et al., 2017] use the WELDON algorithm to detect metastases. Only global annotation is used. The results are on par with classification / detection algorithms trained on detailed annotation.

Overview

1 Motivation

2 Strategies

3 Weak supervision

4 Learning from simulated data

5 Conclusion

6 References

Simulation for the generation of training data

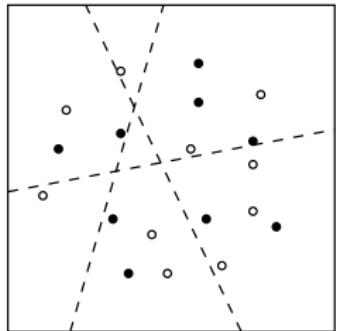
- With simulation, we can generate arbitrary quantities of data with known ground truth.
- Simulation is regularly used for benchmarking of methods.
- Usually, simulation provides us with data similar to real world data, but not identical.
- For this reason, the use of simulated data for training neural networks is limited.
- Idea: can we overcome the differences between simulated and real data by explicit algorithmic strategies?

Domains

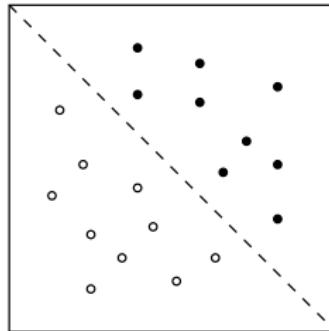
- Domain adaptation: learning a discriminative classifier when training data does not follow the same distribution as the test data.
- Let \mathcal{X} be an input space and $\mathcal{Y} = \{1, \dots, L\}$ the set of L possible labels. We call a domain the distribution over $\mathcal{X} \times \mathcal{Y}$.
- Here we consider two domains: the source domain \mathcal{D}_S and the target domain \mathcal{D}_T .
- The source samples are then drawn from \mathcal{D}_S , the test samples from \mathcal{D}_T^X :

$$\begin{aligned} S &= \{(x_i, y_i)\}_{i=1}^n \sim \mathcal{D}_S^n \\ T &= \{(x_i, y_i)\}_{i=n+1}^N \sim (\mathcal{D}_T^X)^{N-n} \end{aligned}$$

The divergence of domains



(a) Low divergence



(b) High divergence

Figure: Illustration of h-divergence between domains.

- The divergence of domains can be quantified by trying to classify samples according to their source label.
- Hence, we train a classifier from a training set $T = \{(x_i, d_i)\}$:

$$d_i = \begin{cases} 0, & \text{if } x_i \sim \mathcal{D}_S^X \\ 1, & \text{if } x_i \sim \mathcal{D}_T^X \end{cases} \quad (3)$$

- In this case, the h-divergence can be written as $2(1 - 2\epsilon)$, where ϵ is the error of the classifier.

Domain adaptation by adversarial training

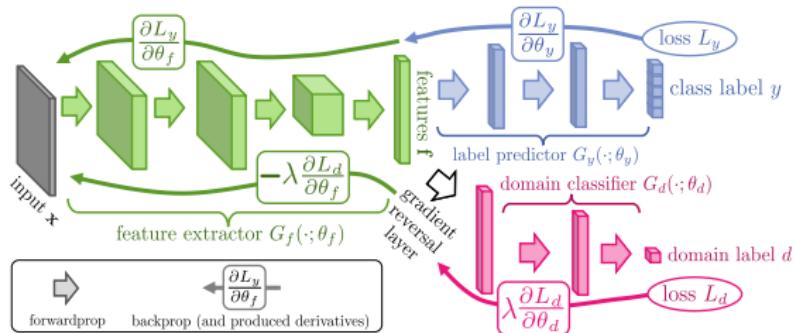


Figure: Domain adaptation by adversarial training. Image taken from [Ganin et al., 2016]

- We seek a representation that provides good prediction results, but low h -divergence.
- This means that this representation (f in the figure) should not allow us to distinguish between source domain and target domain.

Domain adaptation by adversarial training

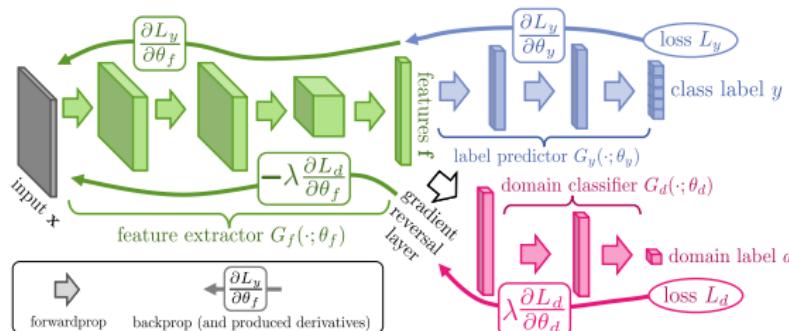


Figure: Domain adaptation by adversarial training. Image taken from [Ganin et al., 2016]

For this we consider the following architecture [Ganin et al., 2016]:

- $G_f(\cdot, \theta_f)$: feature extractor
- $G_y(\cdot, \theta_y)$: label predictor
- $G_d(\cdot, \theta_d)$: domain classifier

Domain adaptation by adversarial training

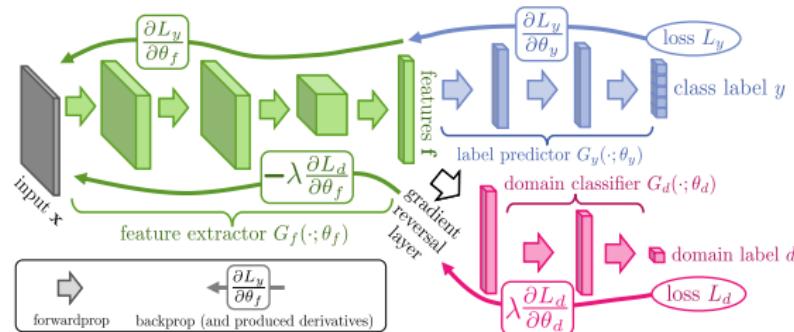


Figure: Domain adaptation by adversarial training. Image taken from [Ganin et al., 2016]

The forward propagation produces then the following loss:

$$L(\theta_f, \theta_y, \theta_d) = L_y(\theta_f, \theta_y) - \lambda L_d(\theta_f, \theta_d) \quad (4)$$

where $L_y(\theta_f, \theta_y)$ is the standard prediction loss for samples from the training set (and thus the source domain), while $L_d(\theta_f, \theta_d)$ is the domain loss, calculated on all samples (labeled samples from the source domain and the unlabeled samples from the target domain).

Domain adaptation by adversarial training - Optimization

- More formally, we write:

$$\begin{aligned} L_y(\theta_f, \theta_y) &= \frac{1}{n} \sum_{i=1}^n L_y(G_y(G_f(x_i, \theta_f), \theta_y), y_i) \\ L_d(\theta_f, \theta_d) &= \frac{1}{n} \sum_{i=1}^n L_d(G_d(G_f(x_i, \theta_f), \theta_d), d_i) \\ &\quad + \frac{1}{N-n} \sum_{i=n}^N L_d(G_d(G_f(x_i, \theta_f), \theta_d), d_i) \\ L(\theta_f, \theta_y, \theta_d) &= L_y(\theta_f, \theta_y) - \lambda L_d(\theta_f, \theta_d) \end{aligned}$$

- This loss has to be minimized with respect to some parameters and maximized with respect to other parameters:

$$(\hat{\theta}_f, \hat{\theta}_y) = \arg \min_{\theta_f, \theta_y} L(\theta_f, \theta_y, \hat{\theta}_d)$$

$$\hat{\theta}_d = \arg \max_{\theta_d} L(\hat{\theta}_f, \hat{\theta}_y, \theta_d)$$

Domain adaptation by adversarial training - gradient reversal

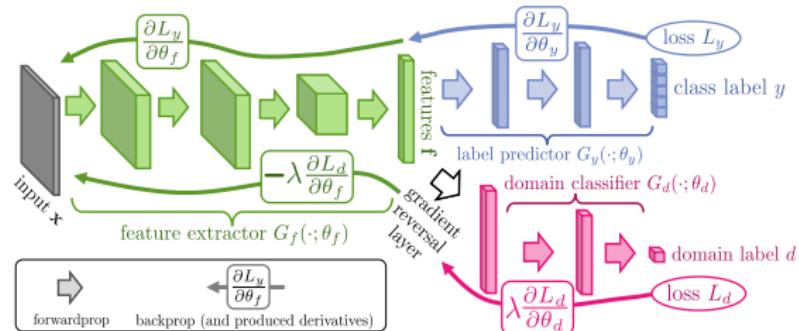


Figure: Domain adaptation by adversarial training. Image taken from [Ganin et al., 2016]

The solution of simultaneous minimization and maximization can be elegantly solved by reversing a gradient layer during back-propagation, as indicated in the figure.

Application example: character classification

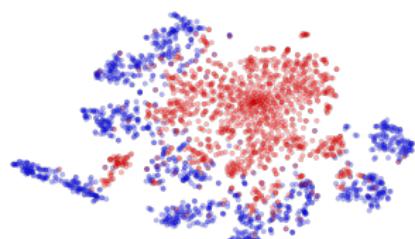
	MNIST	SYN NUMBERS	SVHN	SYN SIGNS
SOURCE				
TARGET				

Figure: Domain adaptation between simulated and real data. MNIST: hand-written digits, MNIST-M: MNIST on top of random image patches, SYN Numbers: simulation created by varying Windows fonts, SVHN: street view house numbers, Image taken from [Ganin et al., 2016]

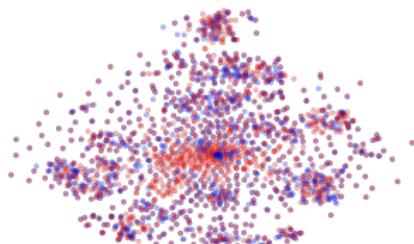
Results indicate that the achievable improvements can reach up to 20% in cases where the simulations are relatively far from the real-world examples.

Application example: character classification

MNIST \rightarrow MNIST-M: top feature extractor layer

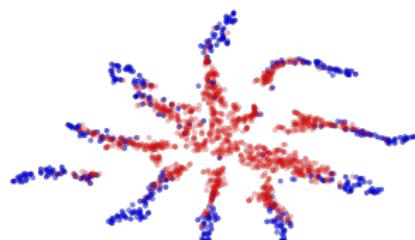


(a) Non-adapted

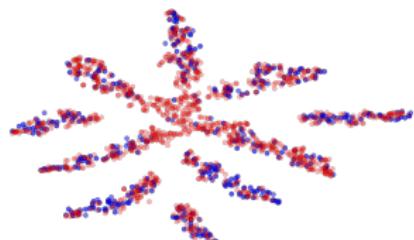


(b) Adapted

SYN NUMBERS \rightarrow SVHN: last hidden layer of the label predictor



(a) Non-adapted



(b) Adapted

Figure: Learned representations with and without domain adaptation. Domains are colored in red and blue. Image taken from [Ganin et al., 2016].

Overview

1 Motivation

2 Strategies

3 Weak supervision

4 Learning from simulated data

5 Conclusion

6 References

Conclusion

- The need for large annotated data sets is currently a bottleneck in many real-world applications of deep learning.
- There is a number of strategies to overcome massive manual image annotation.
- Here, we have seen two strategies:
 - 1 Learning from weakly supervised data
 - 2 Learning from simulations
- Usually, the way in which we setup the annotation strategy usually also influences the methodological developments.

References I

- [Christiansen et al., 2018] Christiansen, E. M., Yang, S. J., Ando, D. M., Javaherian, A., Skibinski, G., Lipnick, S., Mount, E., O’Neil, A., Shah, K., Lee, A. K., Goyal, P., Fedus, W., Poplin, R., Esteva, A., Berndl, M., Rubin, L. L., Nelson, P., and Finkbeiner, S. (2018). In Silico Labeling: Predicting Fluorescent Labels in Unlabeled Images. *Cell*, 173(3):792–803.e19.
- [Courtiol et al., 2017] Courtiol, P., Tramel, E. W., Sanselme, M., and Wainrib, G. (2017). Classification and disease localization in histopathology using only global labels: A weakly supervised approach. *CoRR*, pages 1–13.
- [Durand et al., 2016] Durand, T., Thome, N., and Cord, M. (2016). WELDON: Weakly Supervised Learning of Deep Convolutional Neural Networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4743–4752, Las Vegas, NV, USA. IEEE.

References II

- [Ganin et al., 2016] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17(17):1–35.
- [Iandola et al., 2017] Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2017). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size. In *ICLR*, pages 1–13.
- [J. Boyd et al., 2020] J. Boyd, Z. Gouveia, F. Perez, and T. Walter (2020). Experimentally-Generated Ground Truth for Detecting Cell Types in an Image-Based Immunotherapy Screen. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, pages 886–890.

References III

- [Oquab et al., 2015] Oquab, M., Bottou, L., Laptev, I., and Sivic, J. (2015). Is object localization for free? - Weakly-supervised learning with convolutional neural networks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 685–694, Boston, MA, USA. IEEE.
- [Ouyang et al., 2018] Ouyang, W., Aristov, A., Lelek, M., Hao, X., and Zimmer, C. (2018). Deep learning massively accelerates super-resolution localization microscopy. *Nature Biotechnology*, 36:460.
- [Russakovsky et al., 2015] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252.

References IV

- [Sullivan et al., 2018] Sullivan, D. P., Winsnes, C. F., Åkesson, L., Hjelmare, M., Wiking, M., Schutten, R., Campbell, L., Leifsson, H., Rhodes, S., Nordgren, A., Smith, K., Revaz, B., Finnbogason, B., Szantner, A., and Lundberg, E. (2018). Deep learning is combined with massive-scale citizen science to improve large-scale image classification. *Nature Biotechnology*, 36(9):820–828.
- [Zhou, 2018] Zhou, Z.-H. (2018). A brief introduction to weakly supervised learning. *National Science Review*, 5(1):44–53.