Introduction to Deep Learning
Part III: Practical application

June 16, 2021

Auliya Fitri, Jakob Gawlikowski

Machine Learning Group
Institute of Data Science (DW)
Jena

Curriculum

A: Theoretical introduction – Morning
I. Introduction and basics
II. Advanced concepts
III. Practical application

B. Hands-on seminar – Afternoon
Run prepared Jupyter Notebooks online on Binder, or locally on your own laptop.

Curriculum

III. Practical application

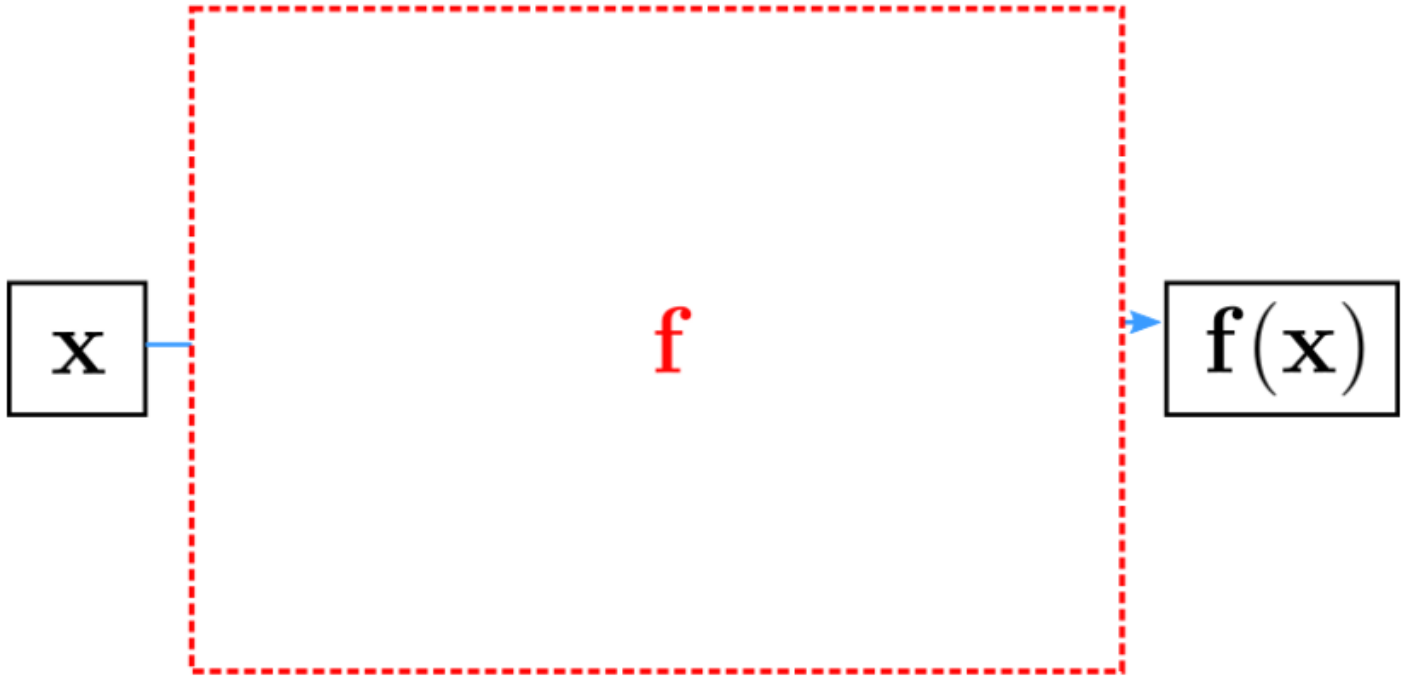
- Frameworks
- Pre-trained models
- Code & knowledge sources
- (Current research topics)

Inspired by lectures from MIT; images taken from these, if not noted otherwise

Frameworks

Frameworks

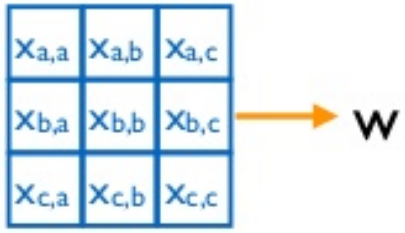
What do DL frameworks commonly offer?



Neural network = parametrized, non-linear function

Frameworks
What do DL frameworks commonly offer?

Tensor algebra



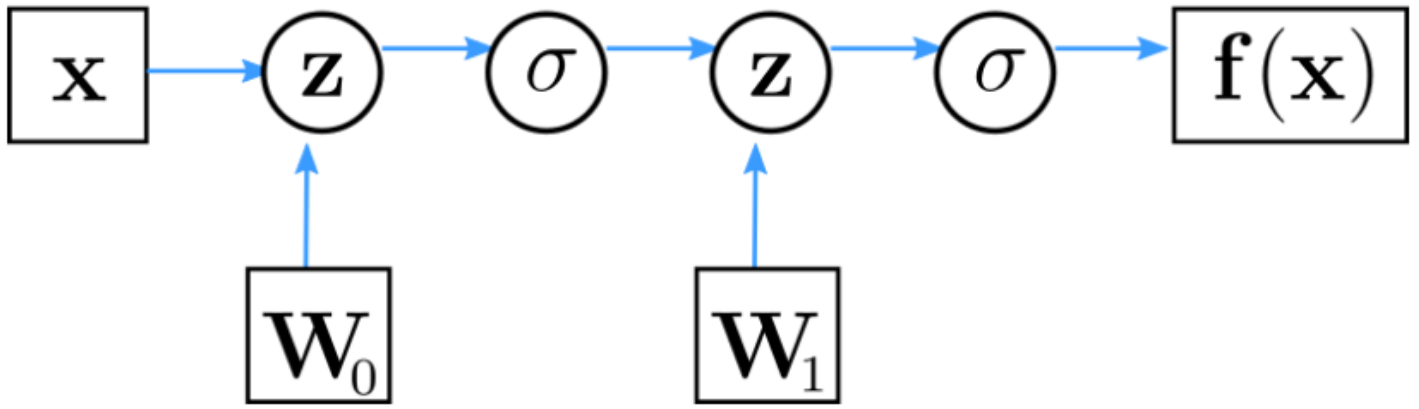
Variable(<initial-value>,
name=<optional-name>)

```
import tensorflow as tf
w = tf.Variable(tf.random_normal([3, 3]), name='w')
y = tf.matmul(x, w)
relu_out = tf.nn.relu(y)
```

https://www.slideshare.net/tw_dsconf/tensorflow-tutorial

Frameworks

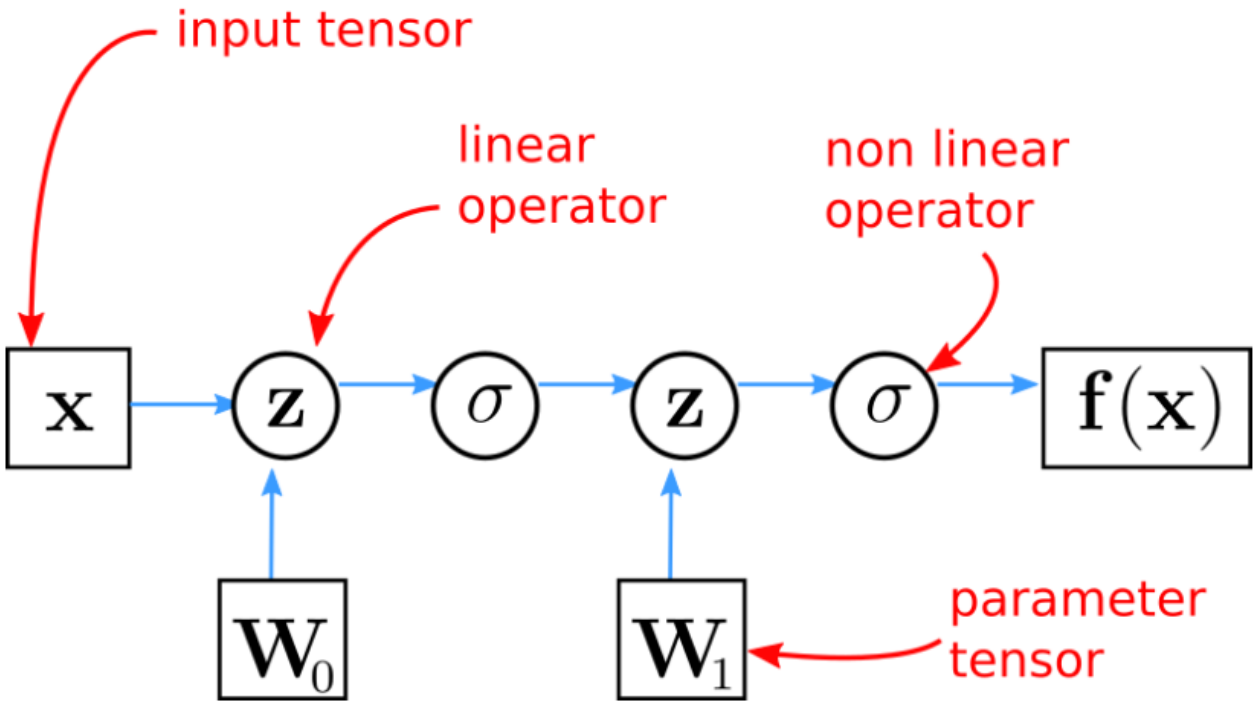
What do DL frameworks commonly offer?



Computation graph: Directed graph of functions, depending on parameters (neuron weights)

Frameworks

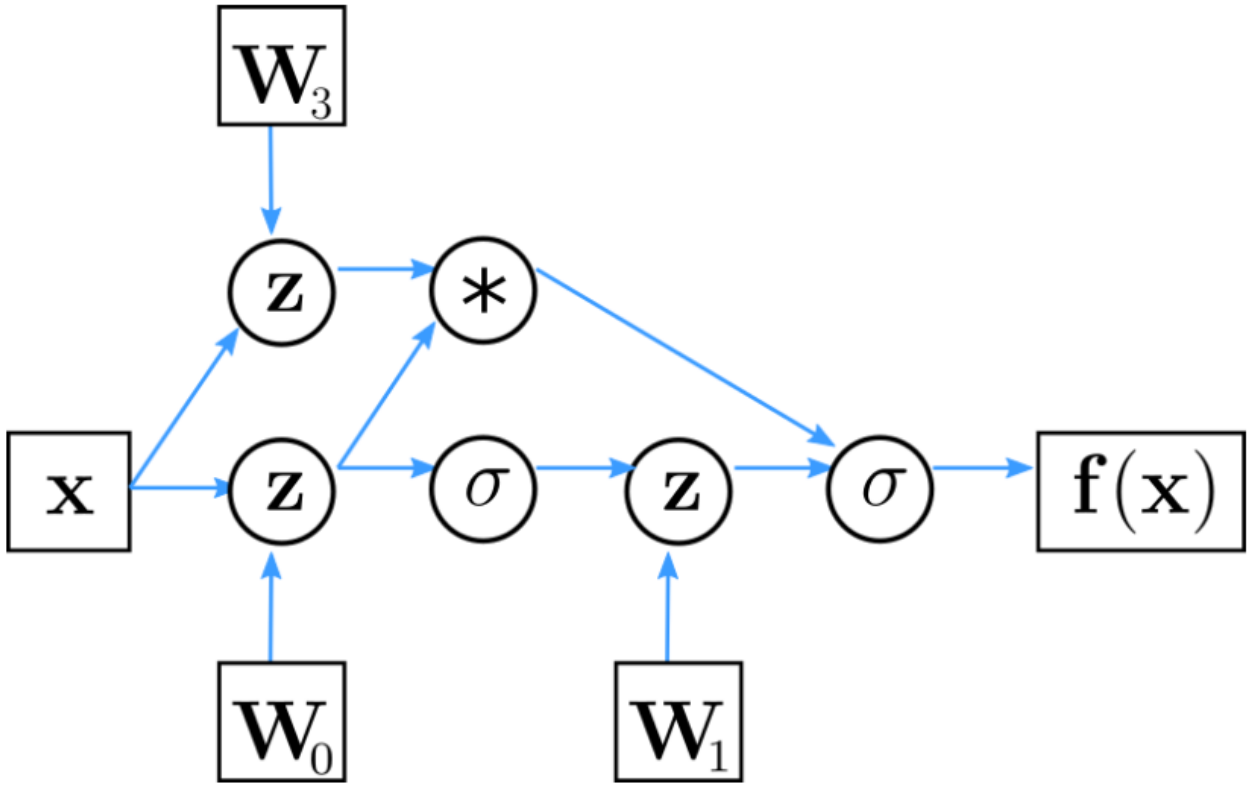
What do DL frameworks commonly offer?



Combination of linear (parametrized) and non-linear functions

Frameworks

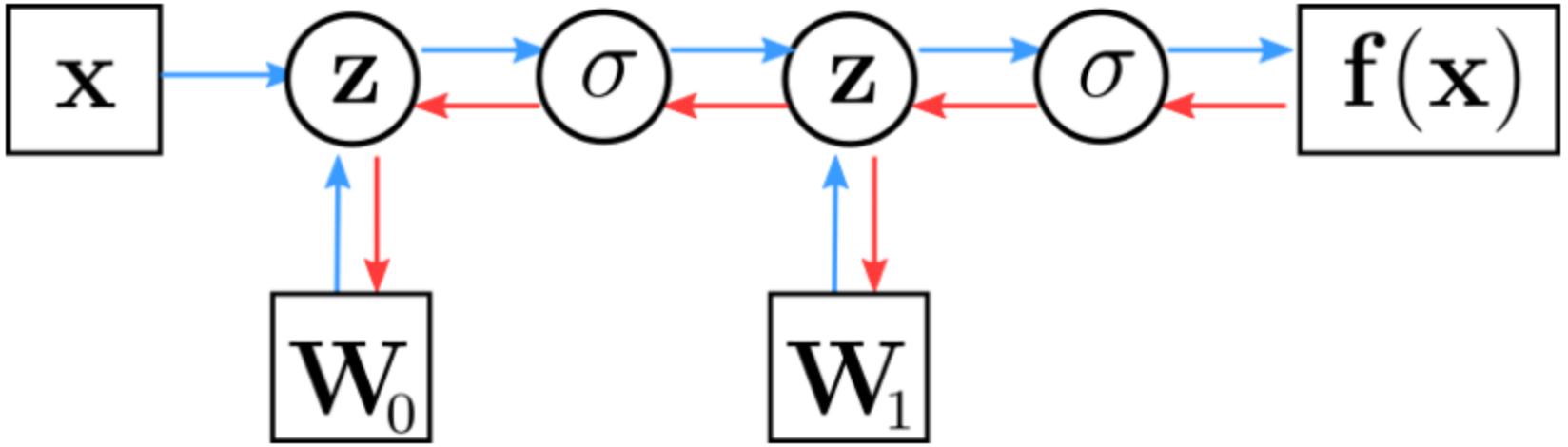
What do DL frameworks commonly offer?



Not only sequential application of functions

Frameworks

What do DL frameworks commonly offer?

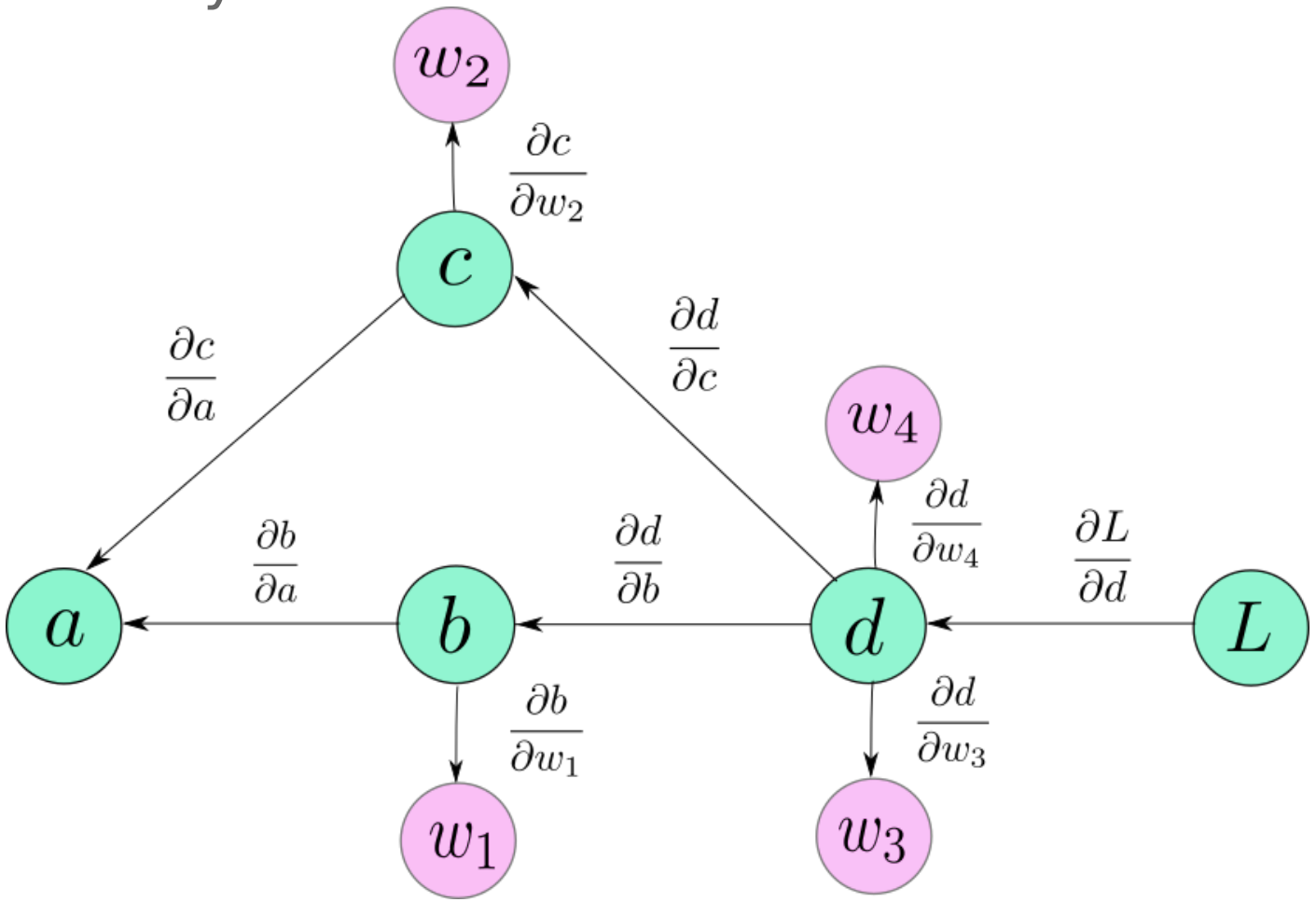


Automatic computation of gradients: all modules are **differentiable**!

Frameworks

What do DL frameworks commonly offer?

Automatic differentiation

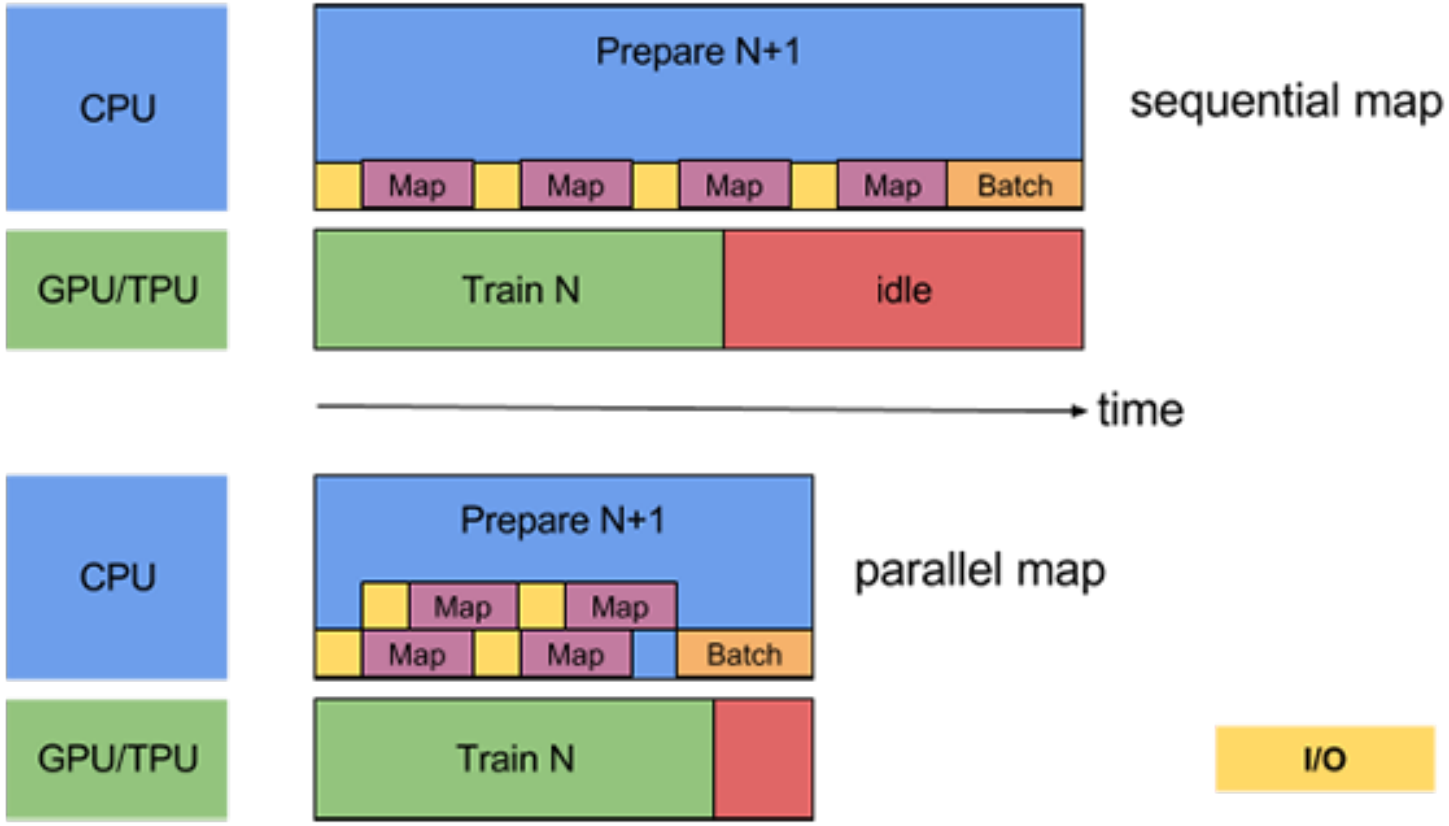


https://towardsdatascience.com/getting-started-with-pytorch-part-1-understanding-how-automatic-differentiation-works-5008282073ec

Frameworks

What do DL frameworks commonly offer?

Algorithmic optimization, Parallelization, Computation on GPU (CUDA)

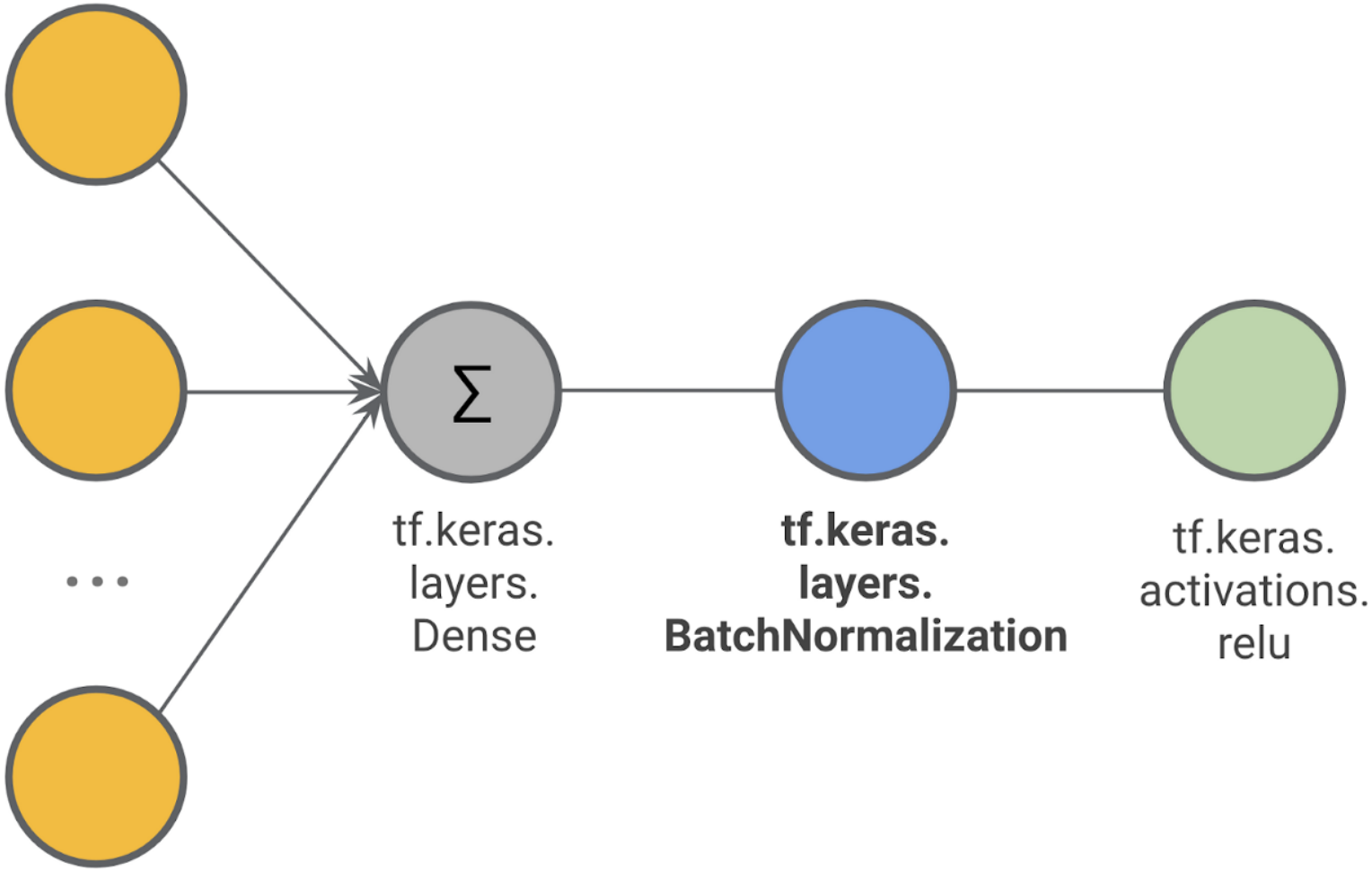


https://www.tensorflow.org/guide/data_performance

Frameworks

What do DL frameworks commonly offer?

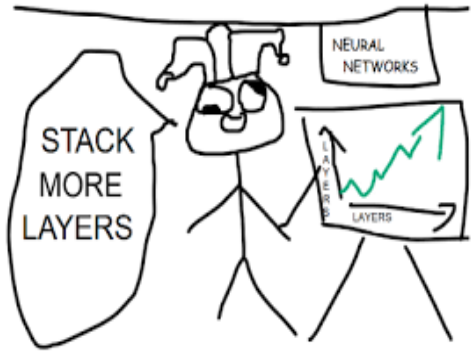
Pre-defined components



https://towardsdatascience.com/how-to-use-batch-normalization-with-tensorflow-and-tf-keras-to-train-deep-neural-networks-faster-60ba4d054b73

Frameworks

What do DL frameworks commonly offer?



Flexibility for new experiments

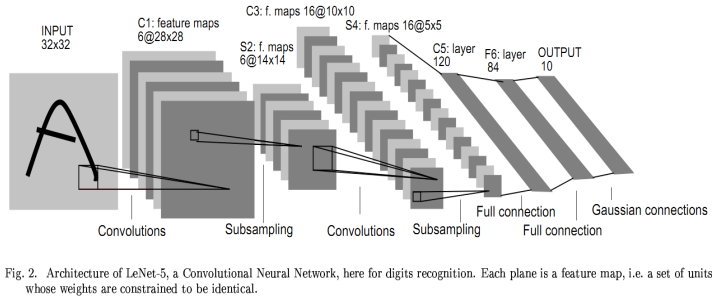


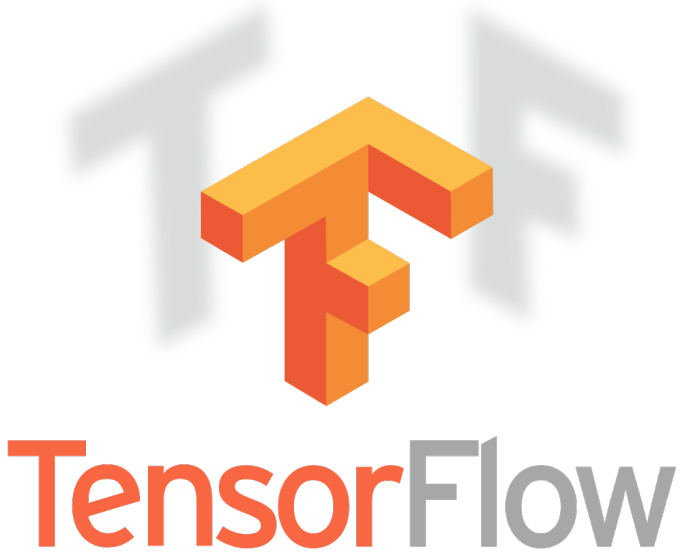
Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Non-linear execution (graphs)

python™

Language usually Python

Frameworks
Available frameworks



TensorFlow

- Developed by Google (Brain)
- Fundamentally: Data stream oriented programming
- Used in in Gmail, Google Photos, Google Maps, Search, Airbnb, ...
- Easy deployment (including mobile)
- Implemented in Python and C
- Supports Python, C/C++, Java, Go, JavaScript, R, Julia, Swift
- Apache License 2.0
- Version 2.0 is much more user-friendly than 1.0

Frameworks
Available frameworks

PyTorch

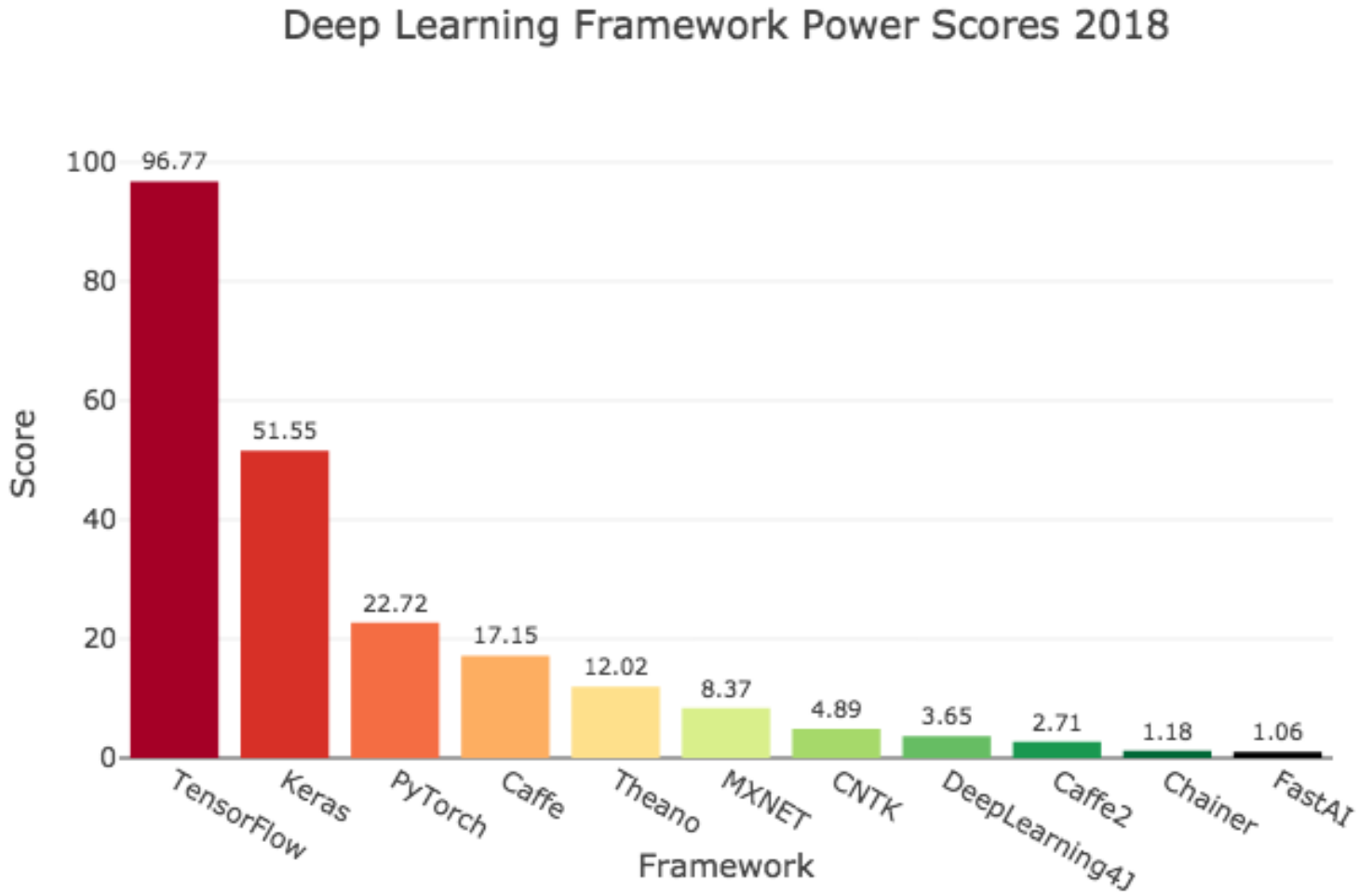
- Developed by Facebook
- Based on Torch (Lua)
- “Autograd” module for differentiation
- Dynamic graph – easy debugging
- User-friendly, easy parallelization
- Used by Facebook, Microsoft, Salesforce, ...
- Implemented in Python and C++
- Supports Python, C++
- BSD license

Frameworks
Available frameworks

Keras

- Open source library by François Chollet
- Further abstraction, simplifies many methods and layers
- Can use various backends
- Now integrated in TensorFlow
- Used by Netflix, Uber, Yelp, ...
- Implemented in Python
- Supports Python, R
- MIT license

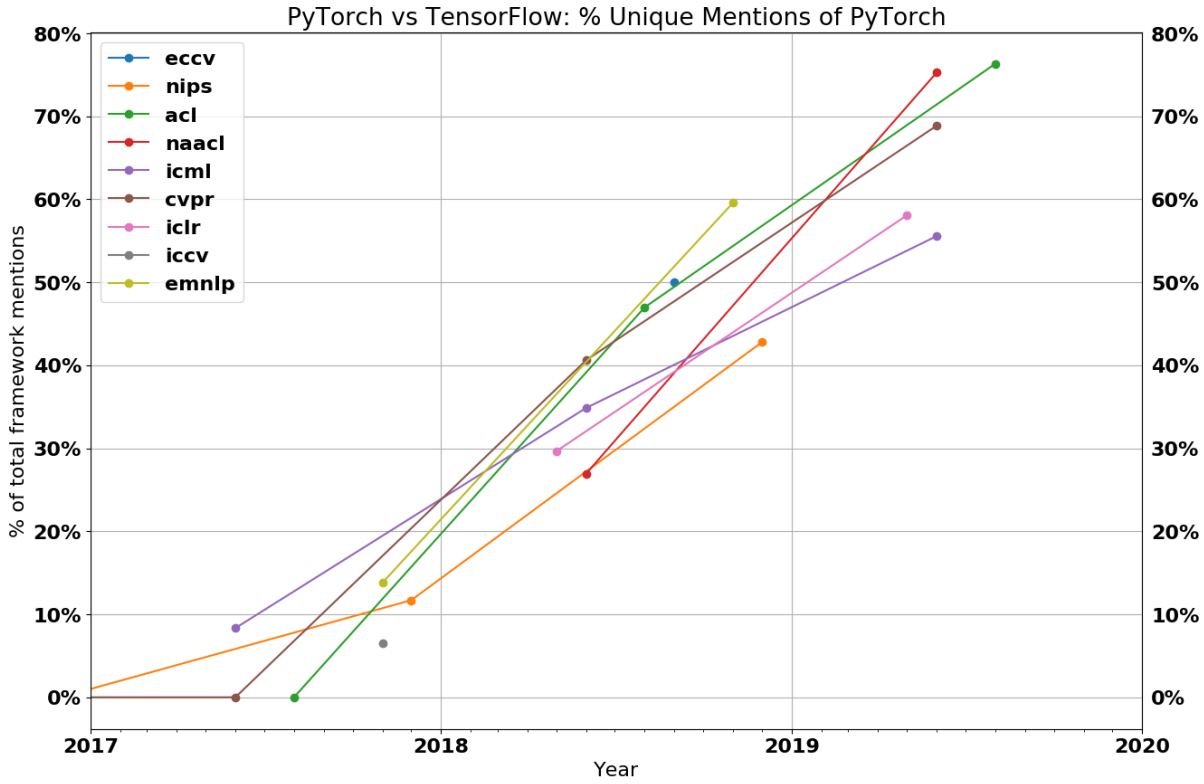
Frameworks
Available frameworks



https://keras.io/why-use-keras/

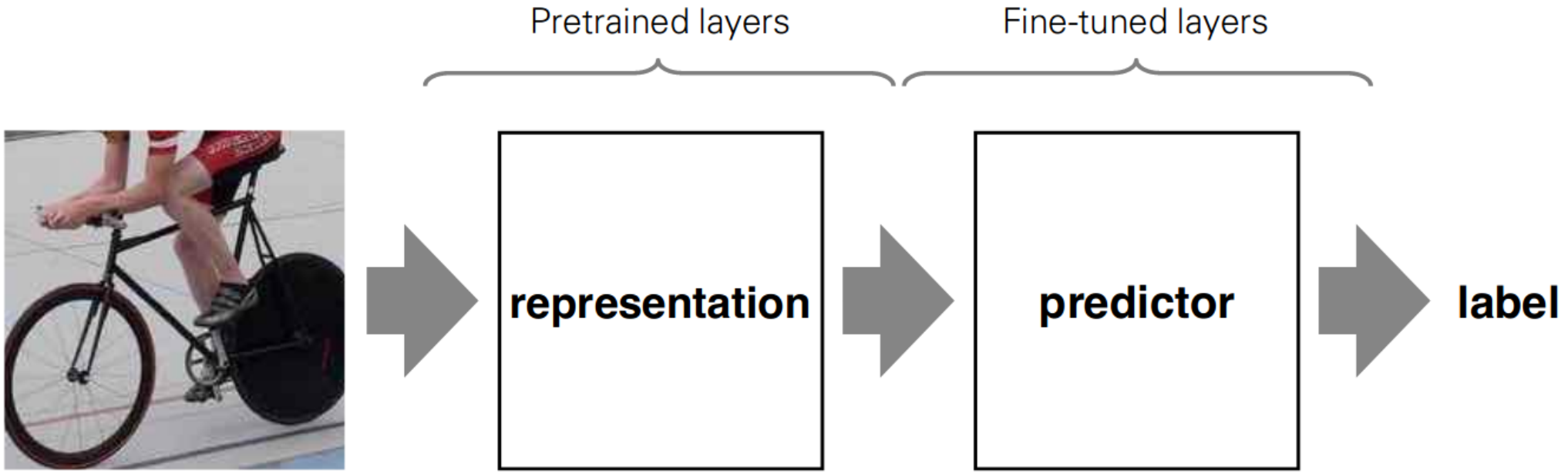
PyTorch’s increasing dominance in research

https://thegradient.pub/state-of-ml-frameworks-2019-pytorch-dominates-research-tensorflow-dominates-industry/



Pre-trained models

Pre-trained models
Transfer learning



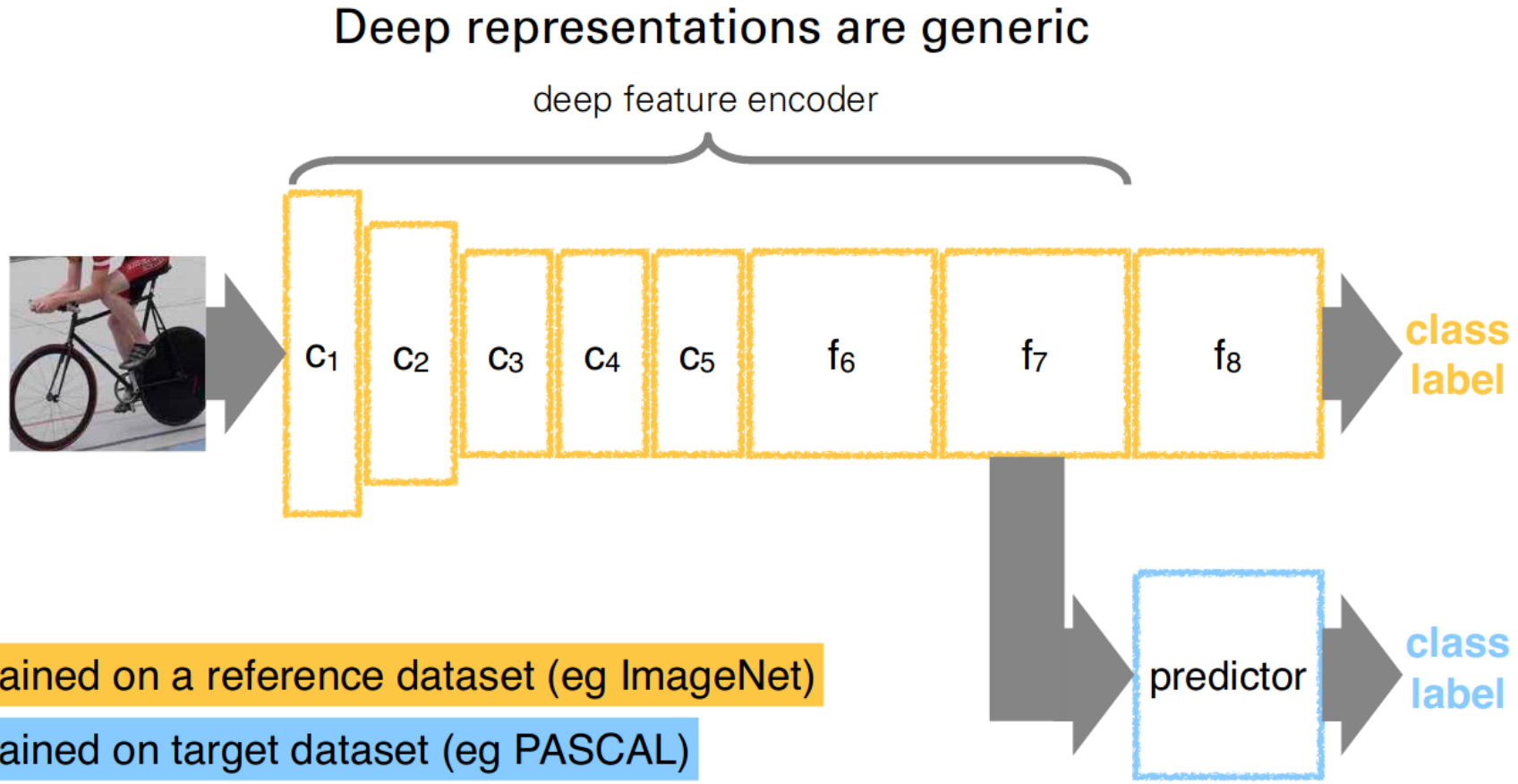
CNN as universal representations

- First several layers in most CNNs are generic
- They can be reused when training data is comparatively scarce.

Application

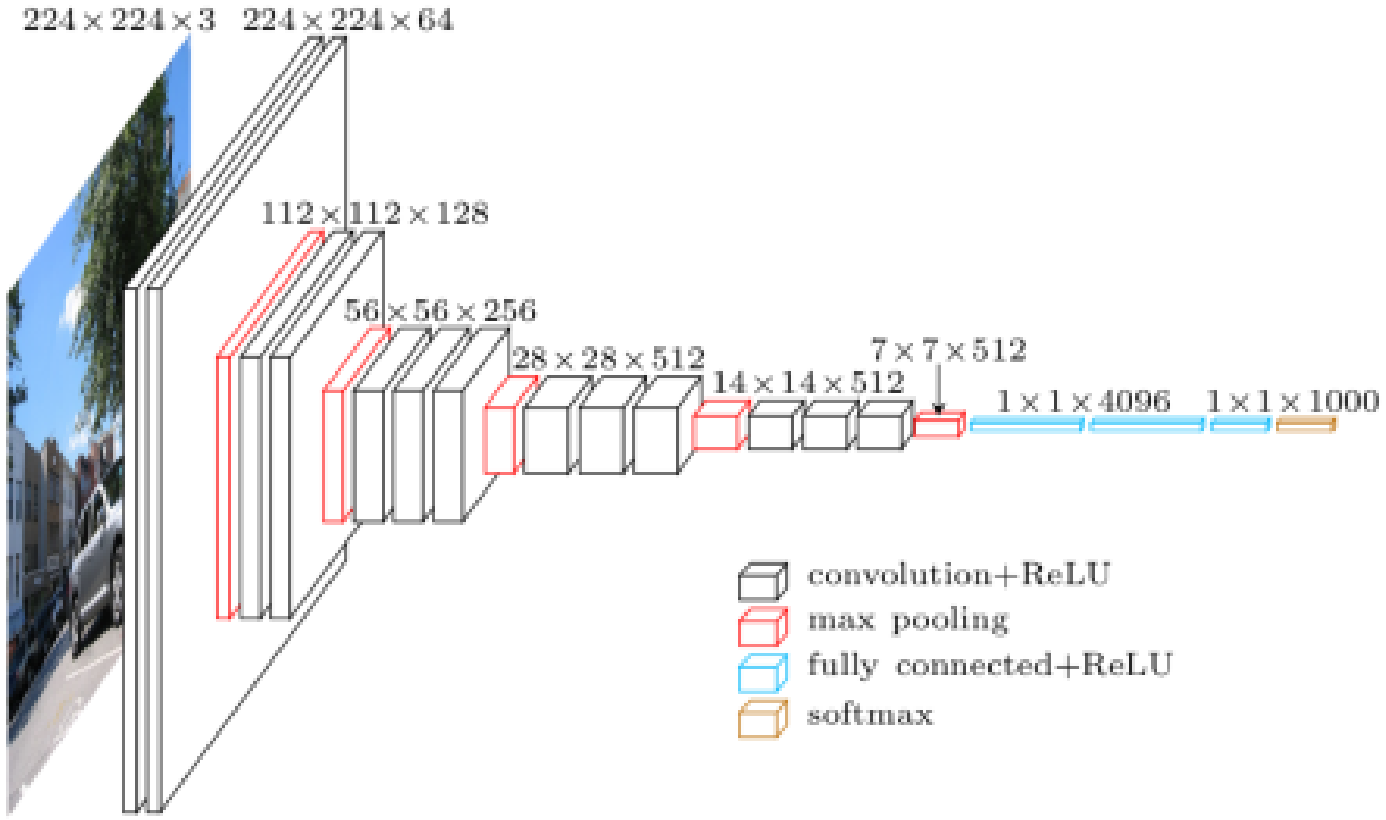
- Pre-train on ImageNet classification 1M images
- Cut at some deep conv or FC layer to get features

Pre-trained models
Transfer learning



- A general purpose deep encoder is obtained by chopping off the last layers of a CNN trained on a large dataset.

Pre-trained models
Examples

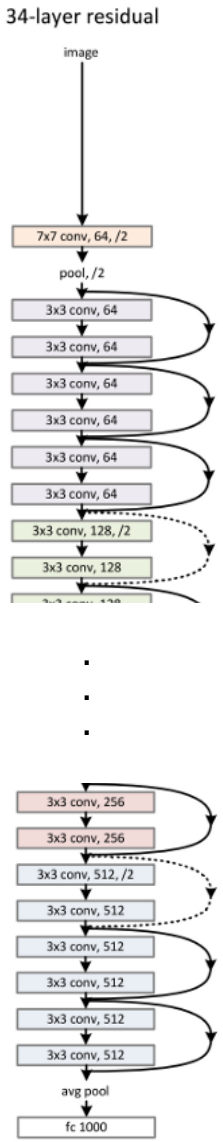


VGG16

- Proposed in 2014
- Used for image classification
- 16 hidden layers, organized in conv blocks
- Later update: VGG19

https://neurohive.io/en/popular-networks/vgg16/

Pre-trained models
Examples

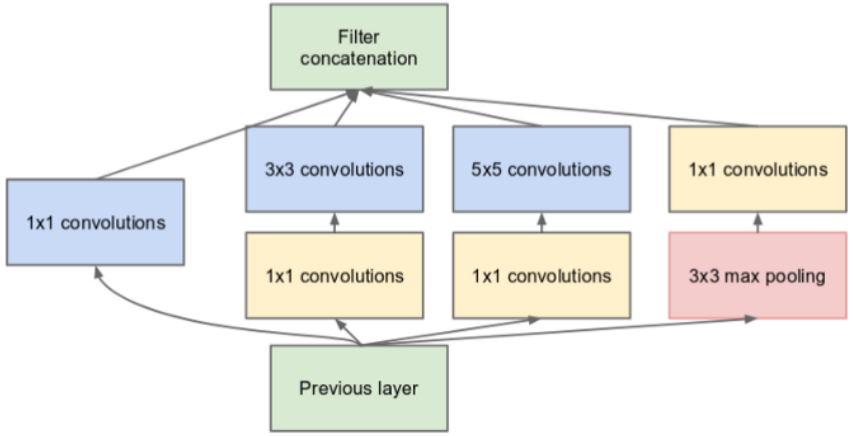


https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035

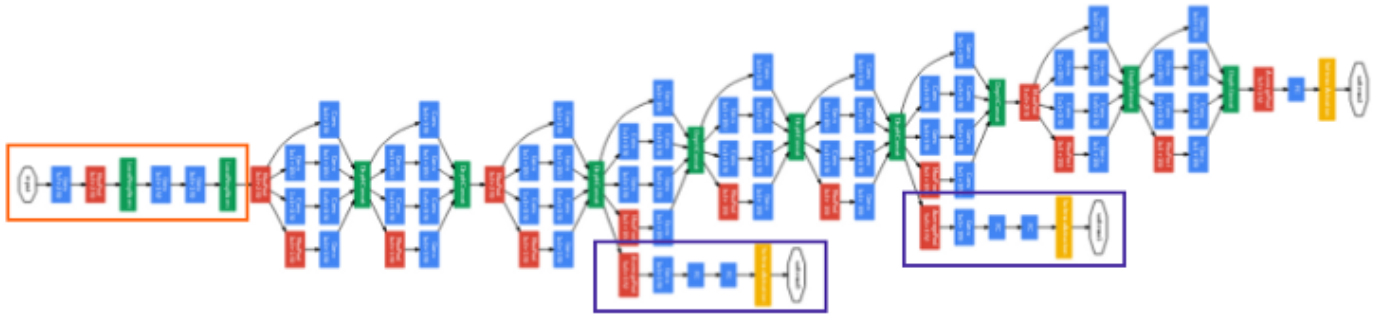
ResNet

- Proposed in 2015
- Used for image classification, object detection, face recognition...
- “Identity shortcut” (or “Highway”) connections
- Stabilizes training of very deep nets
- Many variants

Pre-trained models
Examples



(b) Inception module with dimension reductions



https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202

Inception

- Proposed in 2014
- Used for image classification, object detection, etc.
- Combines filters of various sizes to inception modules – salient features may have different scales
- v1 = GoogLeNet; more versions since then

Pre-trained models
Examples

More examples from computervision:

- LeNet (1998)
- AlexNet (2012)
- Xception (2016)
- ResNeXt (2017)

Other data types:

- Text embeddings (word or sentence level)
- Audio models pre-trained on Google's Audioset

Availability:

- Many already integrated in Frameworks, e.g. Pytorch
- Or found on project pages, GitHub
- Model architectures can be adapted to other tasks (independent of pre-trained weights)

Code & knowledge sources

Code & knowledge source
Code: Github

- Many free implementations of popular and new methods
- Many conferences now require public code for reproducibility
- Usually a good first step for testing and adapting a new method

keras-yolo3

license MIT

Introduction

A Keras implementation of YOLOv3 (Tensorflow backend) inspired by allanzelener/YAD2K.

Quick Start

1. Download YOLOv3 weights from YOLO website.
2. Convert the Darknet YOLO model to a Keras model.
3. Run YOLO detection.

```
wget https://pjreddie.com/media/files/yolov3.weights
python convert.py yolov3.cfg yolov3.weights model_data/yolo.h5
python yolo_video.py [OPTIONS...] --image, for image detection mode, OR
python yolo_video.py [video_path] [output_path (optional)]
```

For Tiny YOLOv3, just do in a similar way, just specify model path and anchor path with `--model model_file` and `--anchors anchor_file`.

Usage

Use --help to see usage of yolo_video.py:

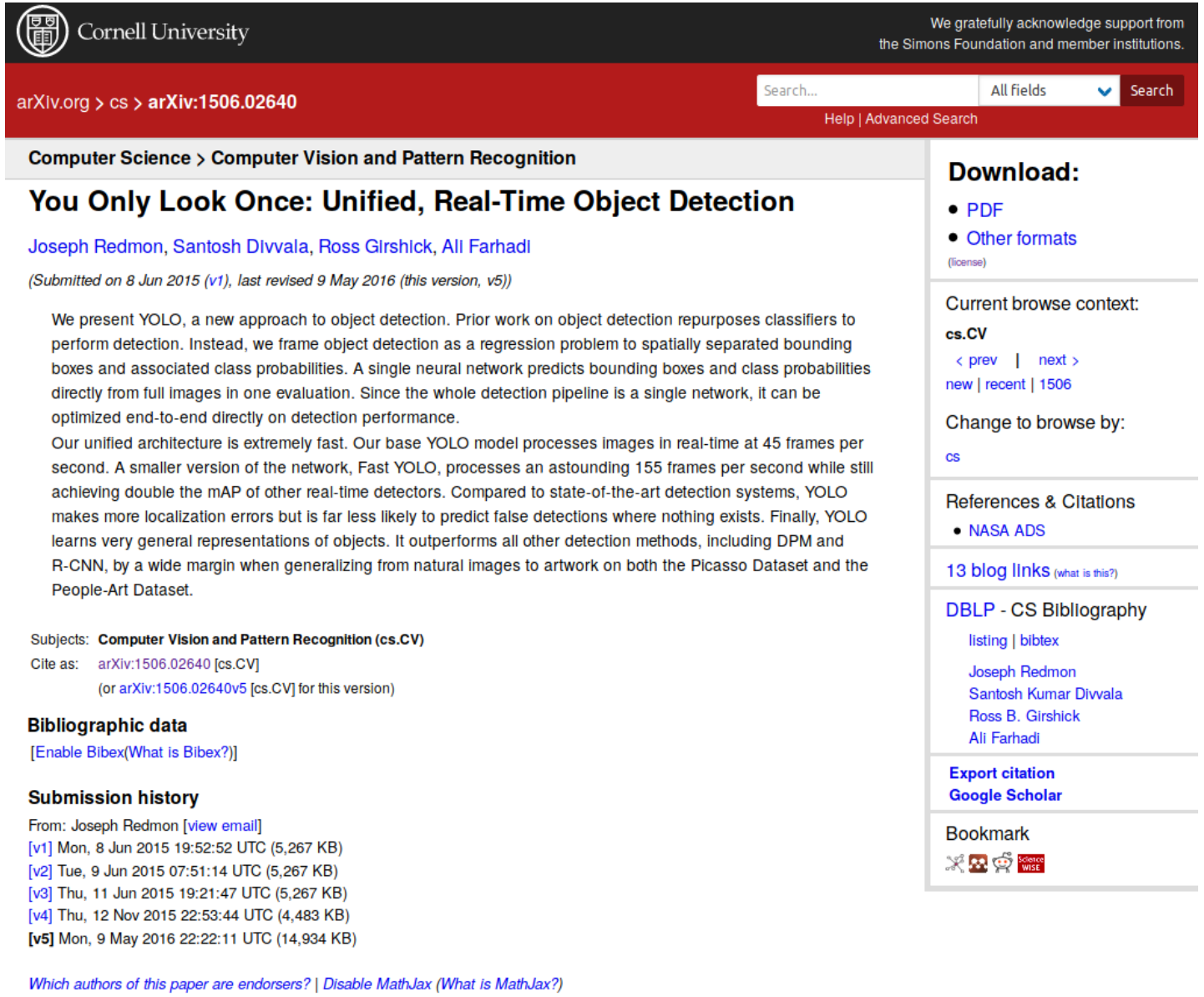
```
usage: yolo_video.py [-h] [--model MODEL] [--anchors ANCHORS]
                     [--classes CLASSES] [--gpu_num GPU_NUM] [--image]
                     [--input] [--output]
```

Code & knowledge sources

Knowledge: arXiv

- Now often the first point of publication for DL papers (sometimes the only one)
- Not peer-reviewed!
- Helpful tool:

http://www.arxiv-sanity.com/



Cornell University

We gratefully acknowledge support from the Simons Foundation and member institutions.

arXiv.org > cs > arXiv:1506.02640

Search... All fields Search

Help | Advanced Search

Computer Science > Computer Vision and Pattern Recognition

You Only Look Once: Unified, Real-Time Object Detection

Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi

(Submitted on 8 Jun 2015 (v1), last revised 9 May 2016 (this version, v5))

We present YOLO, a new approach to object detection. Prior work on object detection repurposes classifiers to perform detection. Instead, we frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance.
Our unified architecture is extremely fast. Our base YOLO model processes images in real-time at 45 frames per second. A smaller version of the network, Fast YOLO, processes an astounding 155 frames per second while still achieving double the mAP of other real-time detectors. Compared to state-of-the-art detection systems, YOLO makes more localization errors but is far less likely to predict false detections where nothing exists. Finally, YOLO learns very general representations of objects. It outperforms all other detection methods, including DPM and R-CNN, by a wide margin when generalizing from natural images to artwork on both the Picasso Dataset and the People-Art Dataset.

Subjects: **Computer Vision and Pattern Recognition (cs.CV)**
Cite as: arXiv:1506.02640 [cs.CV]
(or arXiv:1506.02640v5 [cs.CV] for this version)

Bibliographic data
[Enable Bibex(What is Bibex?)]

Submission history
From: Joseph Redmon [view email]
[v1] Mon, 8 Jun 2015 19:52:52 UTC (5,267 KB)
[v2] Tue, 9 Jun 2015 07:51:14 UTC (5,267 KB)
[v3] Thu, 11 Jun 2015 19:21:47 UTC (5,267 KB)
[v4] Thu, 12 Nov 2015 22:53:44 UTC (4,483 KB)
[v5] Mon, 9 May 2016 22:22:11 UTC (14,934 KB)

Which authors of this paper are endorsers? | Disable MathJax (What is MathJax?)

Download:
- PDF
- Other formats (license)

Current browse context: **cs.CV**
< prev | next >
new | recent | 1506

Change to browse by: cs

References & Citations
- NASA ADS

13 blog links (what is this?)

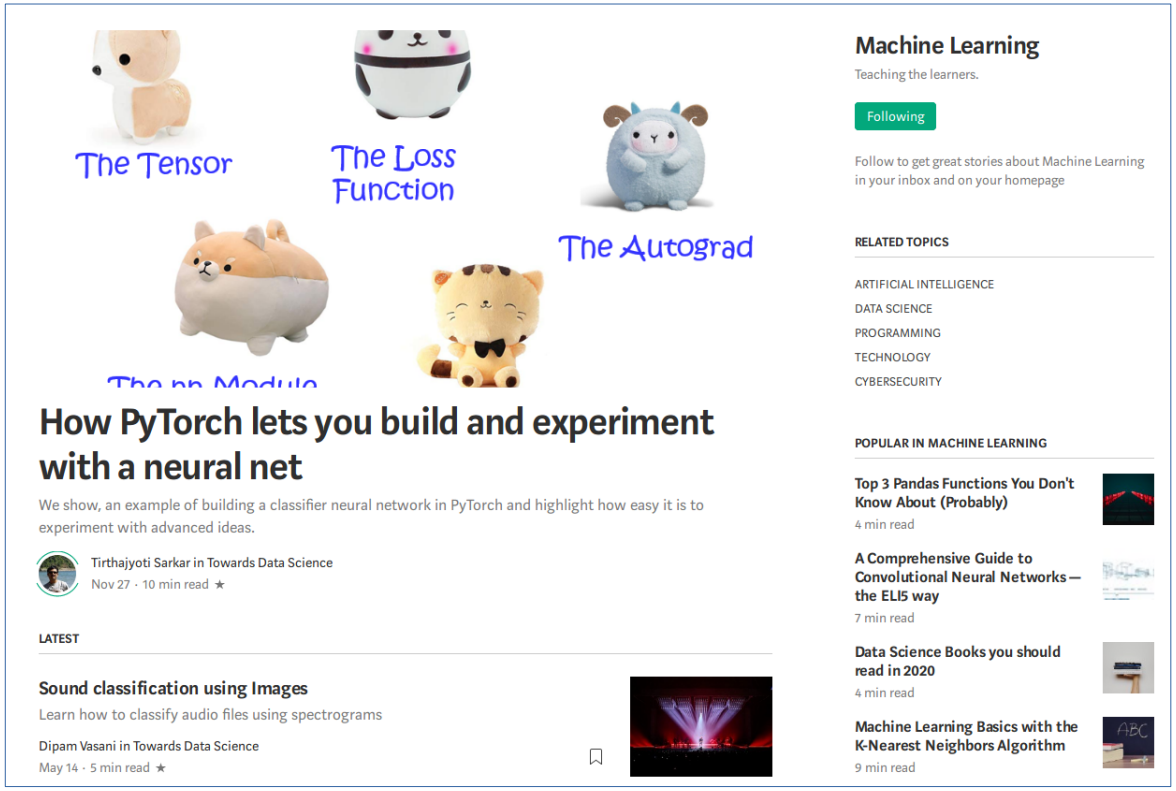
DBLP - CS Bibliography
listing | bibtex
Joseph Redmon
Santosh Kumar Divvala
Ross B. Girshick
Ali Farhadi

Export citation
Google Scholar

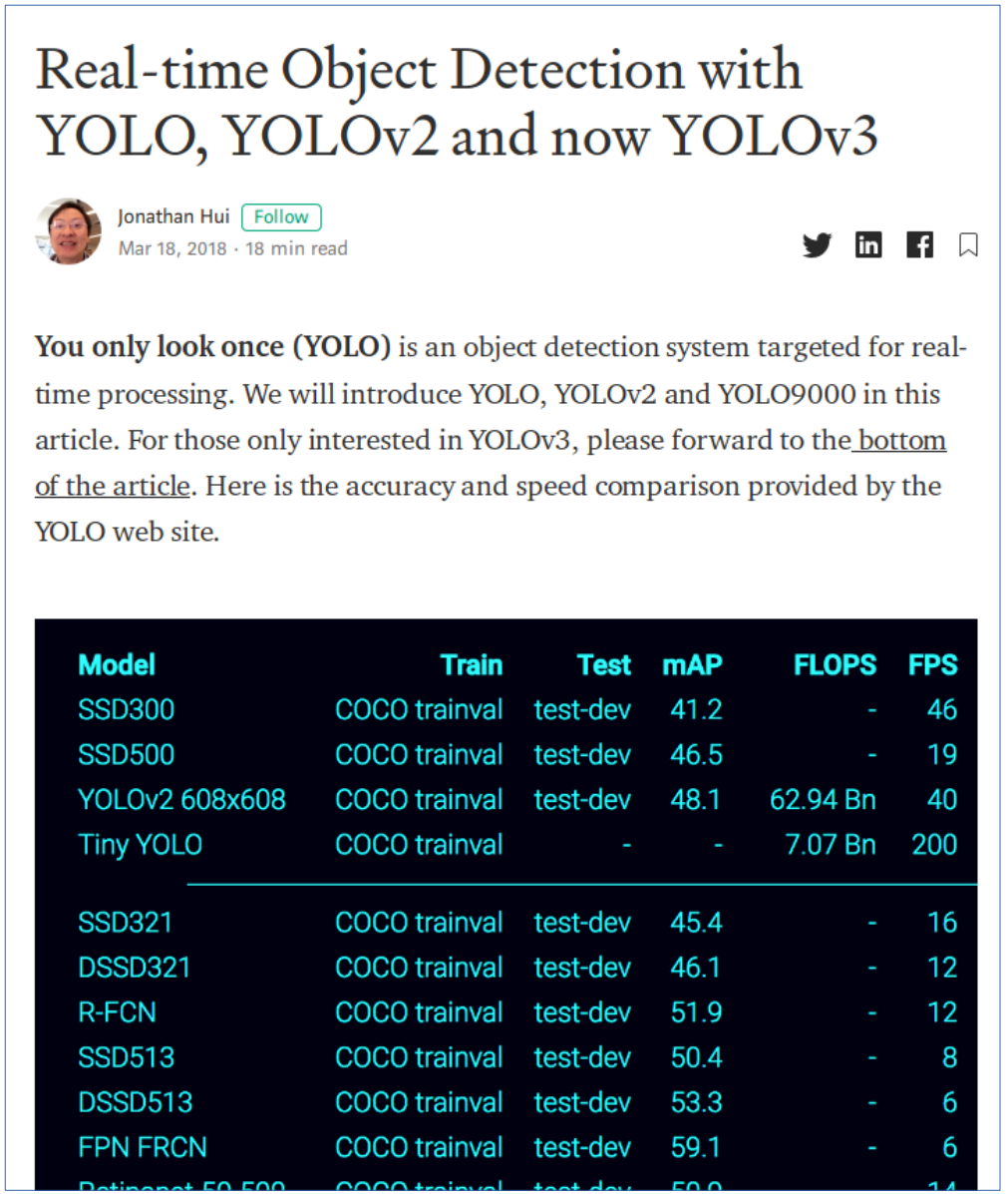
Bookmark

Code & knowledge sources

Knowledge: Blogs



https://medium.com/topic/artificial-intelligence
https://medium.com/topic/machine-learning



Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3

Jonathan Hui Follow
Mar 18, 2018 · 18 min read

You only look once (YOLO) is an object detection system targeted for real-time processing. We will introduce YOLO, YOLOv2 and YOLO9000 in this article. For those only interested in YOLOv3, please forward to the bottom of the article. Here is the accuracy and speed comparison provided by the YOLO web site.

Model	Train	Test	mAP	FLOPS	FPS
SSD300	COCO trainval	test-dev	41.2	-	46
SSD500	COCO trainval	test-dev	46.5	-	19
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40
Tiny YOLO	COCO trainval	-	-	7.07 Bn	200
SSD321	COCO trainval	test-dev	45.4	-	16
DSSD321	COCO trainval	test-dev	46.1	-	12
R-FCN	COCO trainval	test-dev	51.9	-	12
SSD513	COCO trainval	test-dev	50.4	-	8
DSSD513	COCO trainval	test-dev	53.3	-	6
FPN FRCN	COCO trainval	test-dev	59.1	-	6

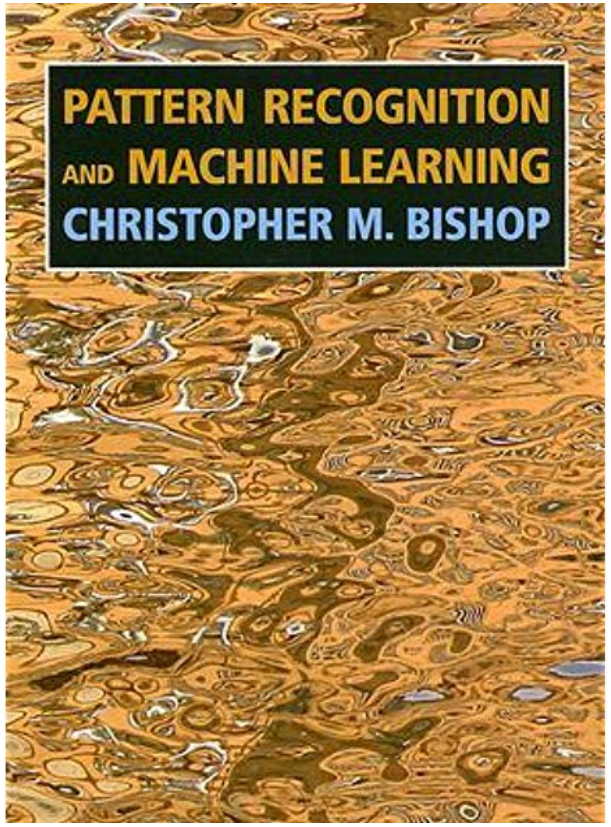
Code & knowledge sources

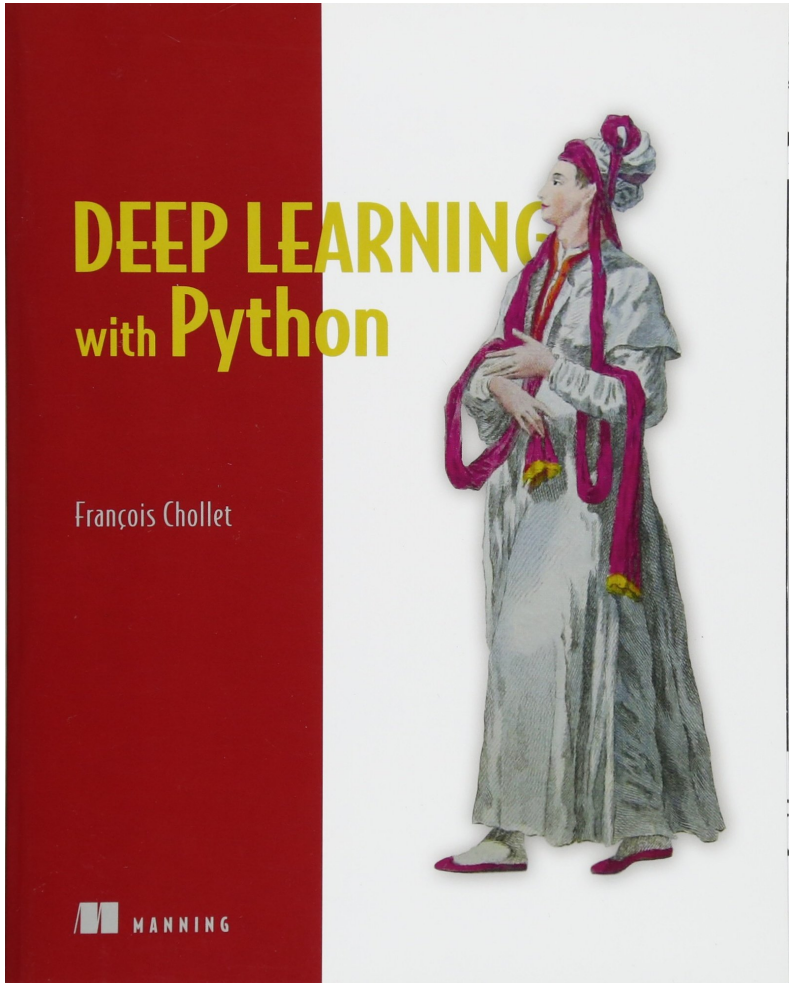
Knowledge: Conferences (selection)

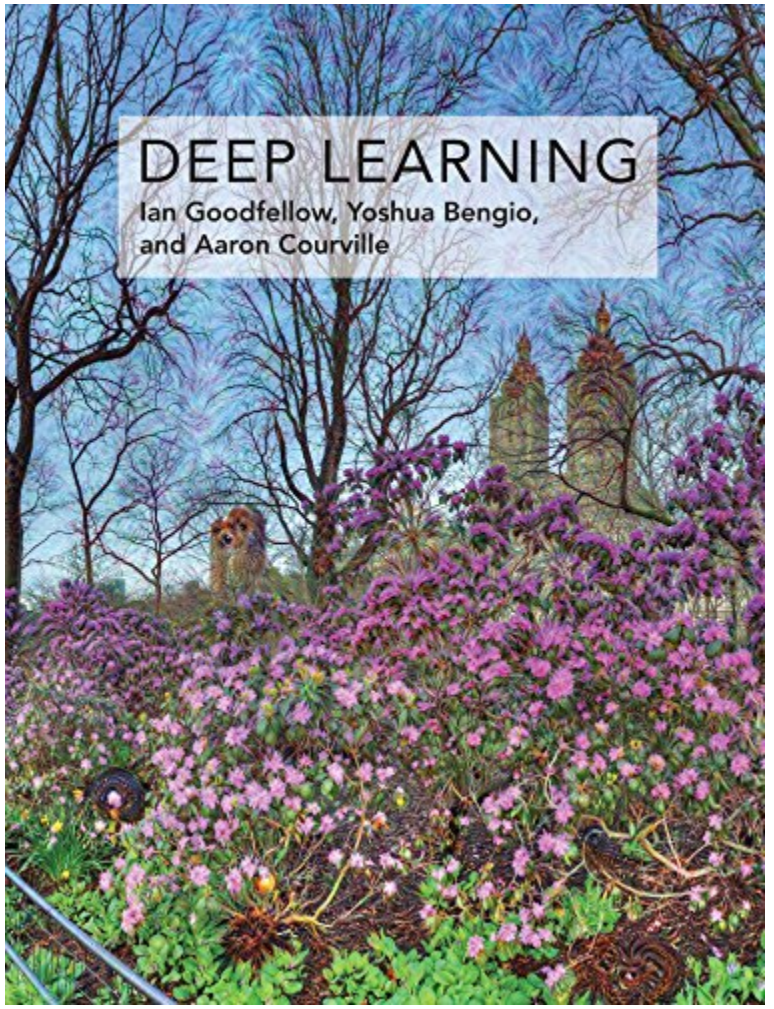
...and of course many domain-specific conferences

Code & knowledge sources

Knowledge: Books



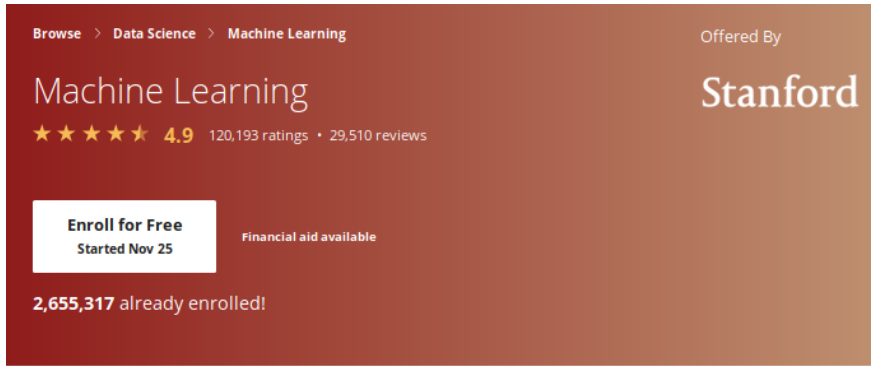


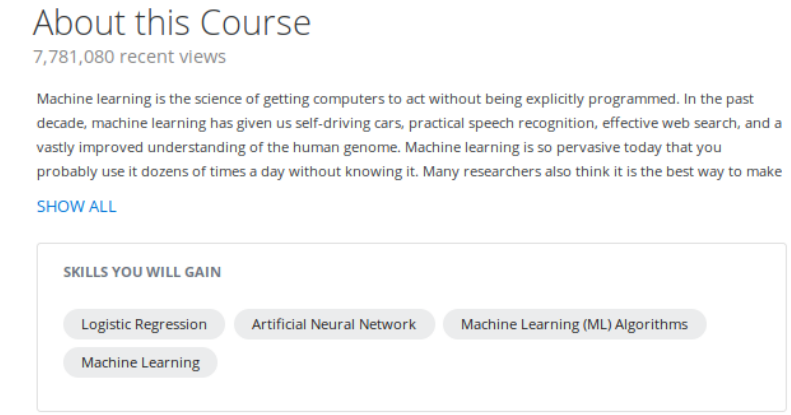


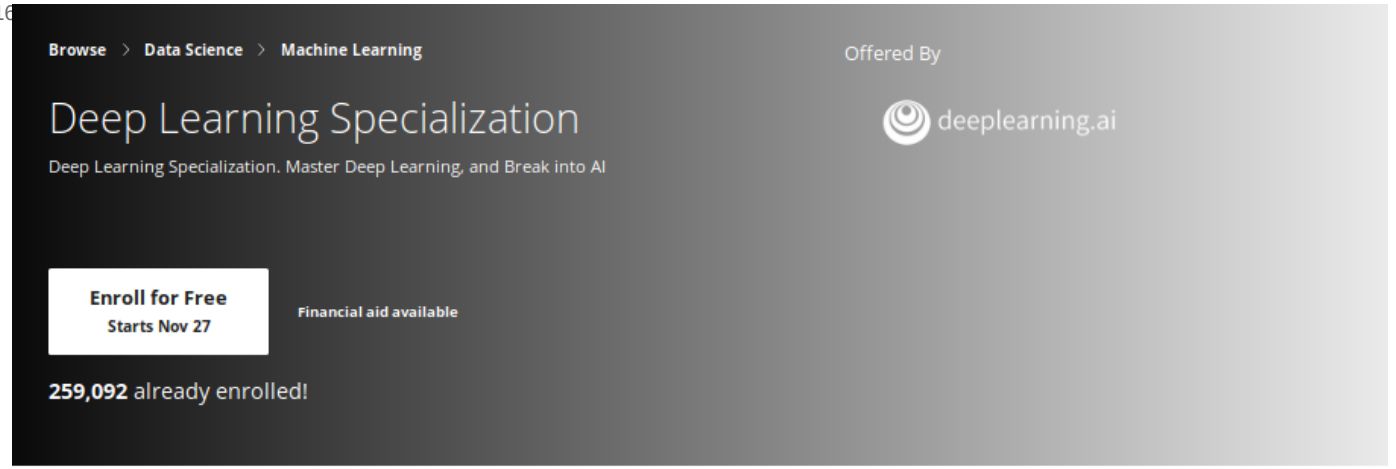
http://
www.deeplearningbook.org/

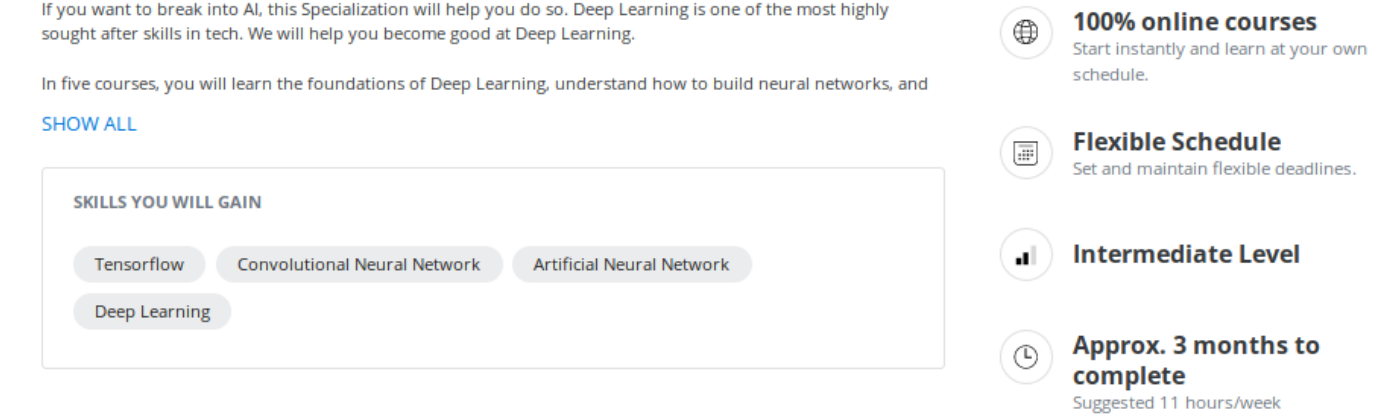
Code & knowledge sources

Knowledge: Online courses









https://www.coursera.org/

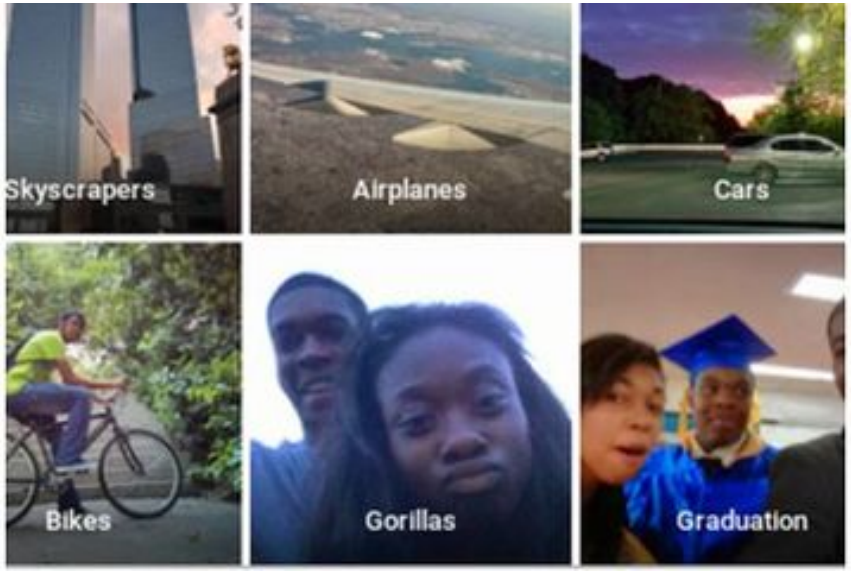
Current research topics

Combining Different Modalities

Facebook‘s Hateful Memes Challange (results presented today at Neurips)

https://ai.facebook.com/blog/hateful-memes-challenge-and-data-set/

Current research topics
AI fairness and transparency

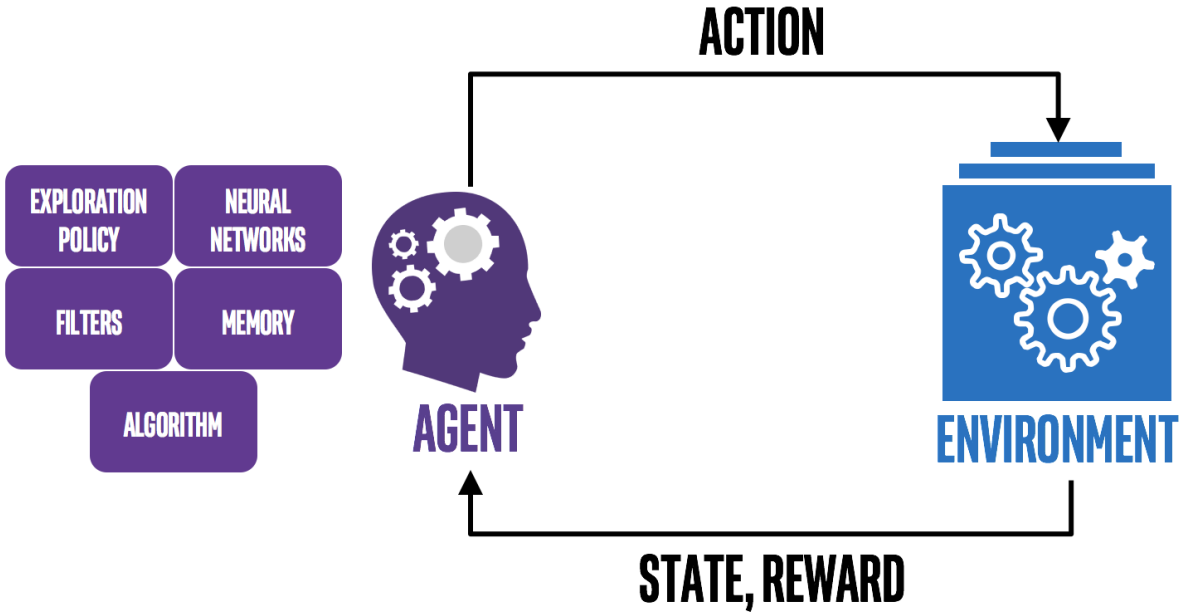


https://www.hrkatha.com/recruitment/amazon-discreetly-abandoned-gender-biased-ai-based-recruiting-tool/

https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist

https://blogs.wsj.com/digits/2015/07/01/google-mistakenly-tags-black-people-as-gorillas-showing-limits-of-algorithms/

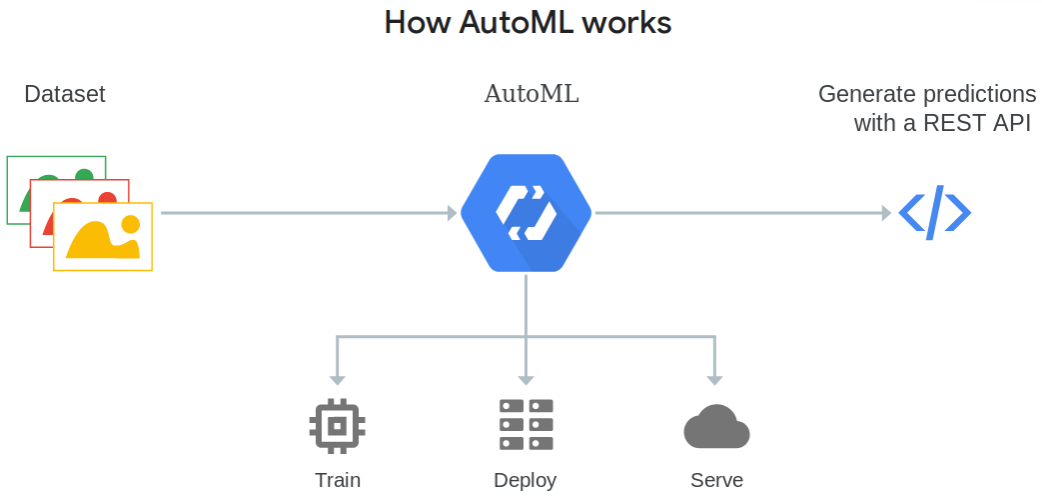
Current research topics
Reinforcement learning



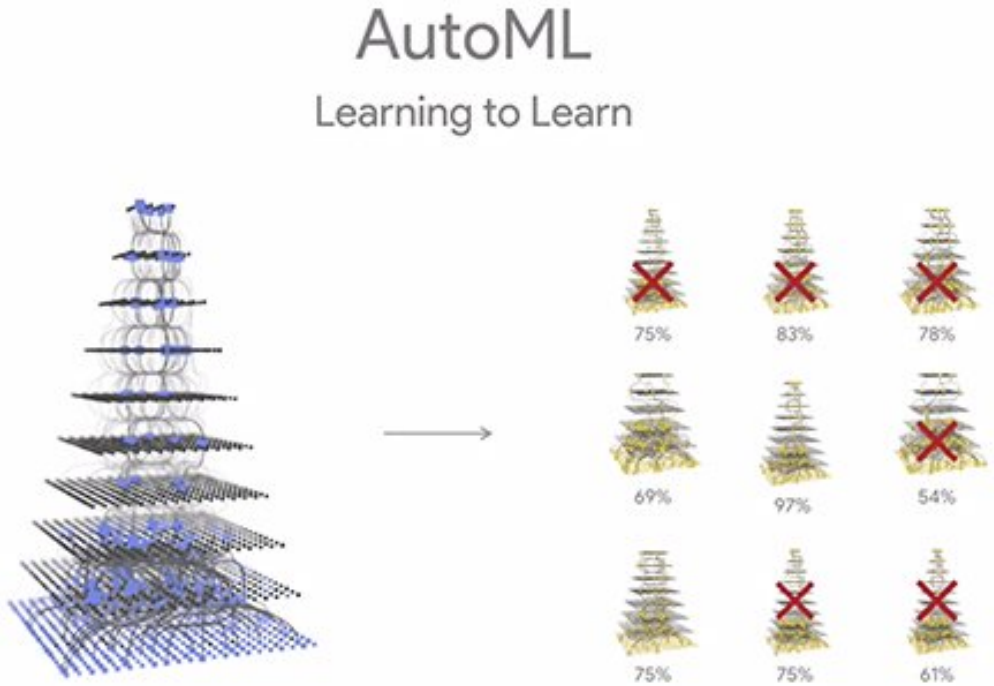
https://robohub.org/deep-learning-in-robotics/

https://medium.com/ai%C2%B3-theory-practice-business/reinforcement-learning-part-1-a-brief-introduction-a53a849771cf

Current research topics
Neural architecture search (AutoML)

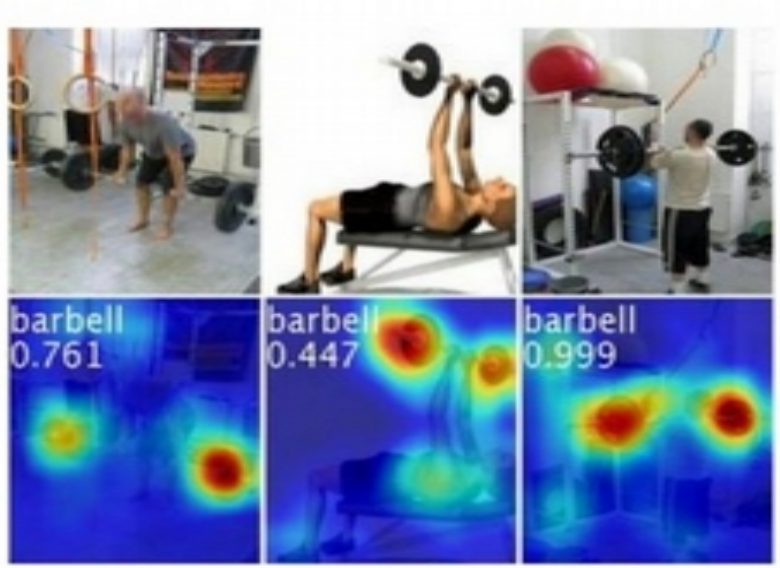


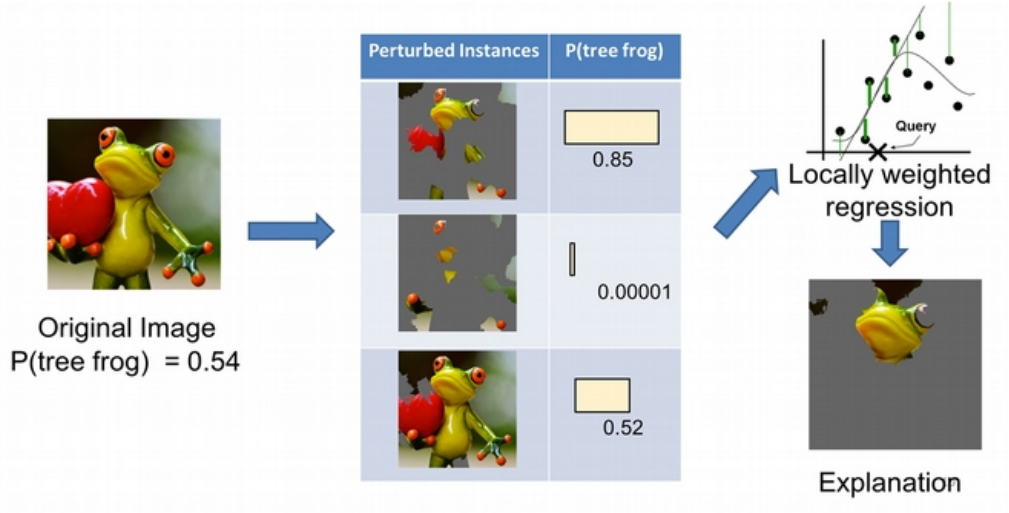
https://cloud.google.com/automl/

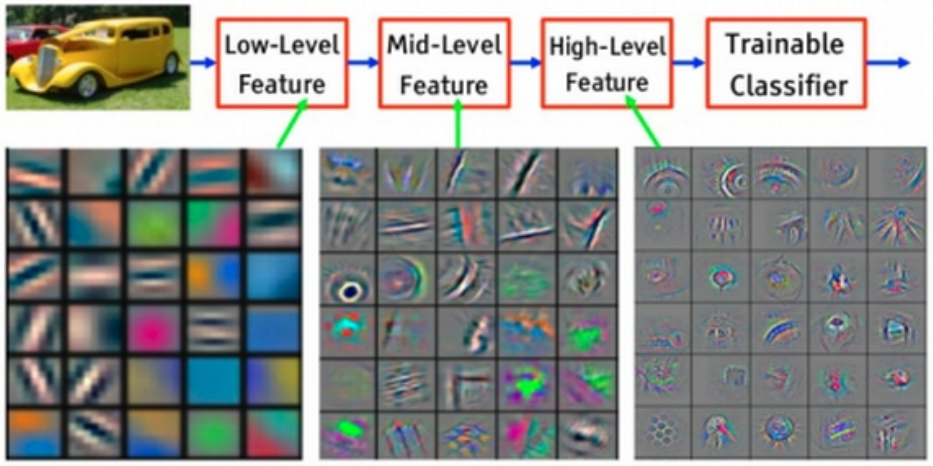


https://www.sciencealert.com/google-is-improving-its-artificial-intelligence-with-artificial-intelligence

Current research topics
Explainability/Interpretability





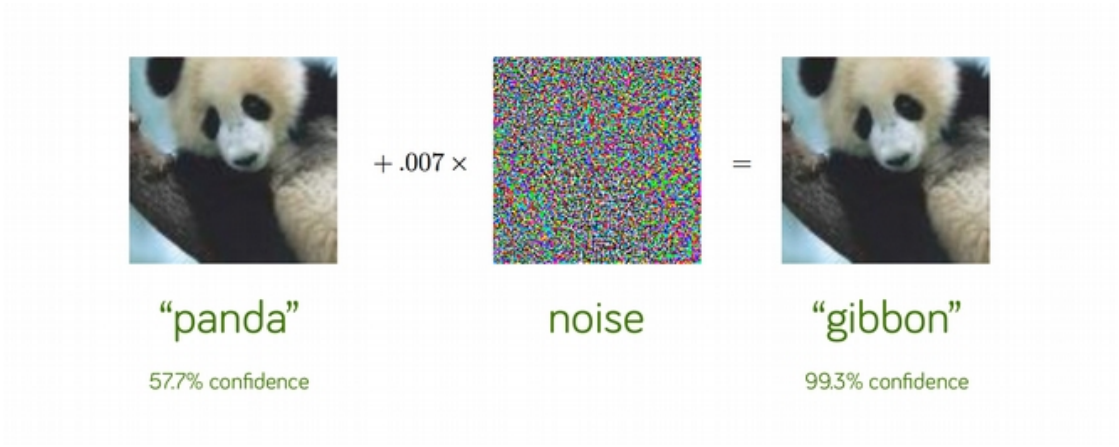


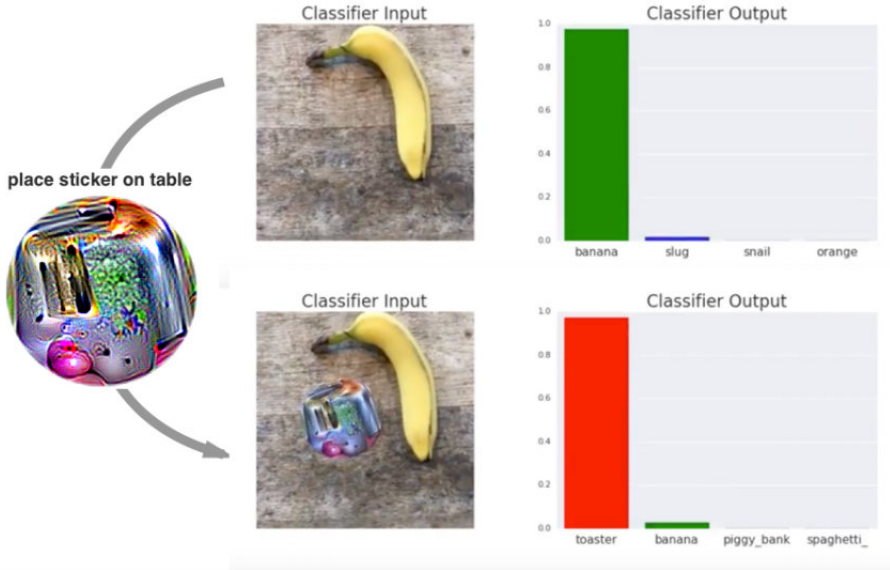
M. T. Ribeiro, S. Singh, C. Guestrin: Local Interpretable Model-Agnostic Explanations (LIME): An Introduction. https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime

B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba: Learning Deep Features for Discriminative Localization. CVPR 2016.

M. D. Zeiler, R. Fergus: Visualizing and Understanding Convolutional Networks. ECCV 2014.

Current research topics
Robustness, uncertainty estimation





I. Goodfellow: NIPS 2016 Tutorial: Generative Adversarial Networks.

II. https://techxplore.com/news/2018-01-world-toaster-ai.html

Thank you for your interest!

auliya.fitri@dlr.de
jakob.gawlikowski@dlr.de

Curriculum

Next: B. Hands-on seminar – Try it out:

- Understand PyTorch‘s Tensor library and neural networks at a high level
- Train a small neural network for
  - regression analysis.
  - classification.
  - image classification.