Data Science Academy

Data Science for Business – Becoming a Data Science Expert (D)

Pilot Presentation:
for participants of and use in the pilot only

13.09.2021

Boehringer Ingelheim

Agenda week three

Introduction

Schedule week three

Week 3			
	Day 1 Monday, 13.09.2021		Day 2 Tuesday, 14.09.2021
Start: 12:00	Recap	Start: 12:00	Recap
	7 – DL: Image Recognition (Part 1)		8 – DL: NLP (Part 1)
14:00 – 15:00	Break	14:00 – 15:00	Break
	7 – DL: Image Recognition (Part 2)		8 – DL: NLP (Part 2)
End: 18:00	Q&A and Feedback	End: 18:00	Q&A and Feedback

We will also have several short coffee breaks in between.

Feedback for pilot training

We aim to provide a great training experience for you and are looking forward to receiving your feedback!

You will have three different ways to give us your feedback on each training day:

1. We will have an **anonymized** feedback collection **after the last session** of each day per **Myforms.**
2. We will have an **open feedback round and discussion** at the **end of each training day.**
3. Please also **take notes** regarding your ideas during the sessions: **locally or via the Mural Board** which you can reach via LINK.

Quiz: Recap Week Two

Please join at slido.com with #031 077.

Let’s go through some questions together.

Let’s see what you think. All answers will be anonymous.

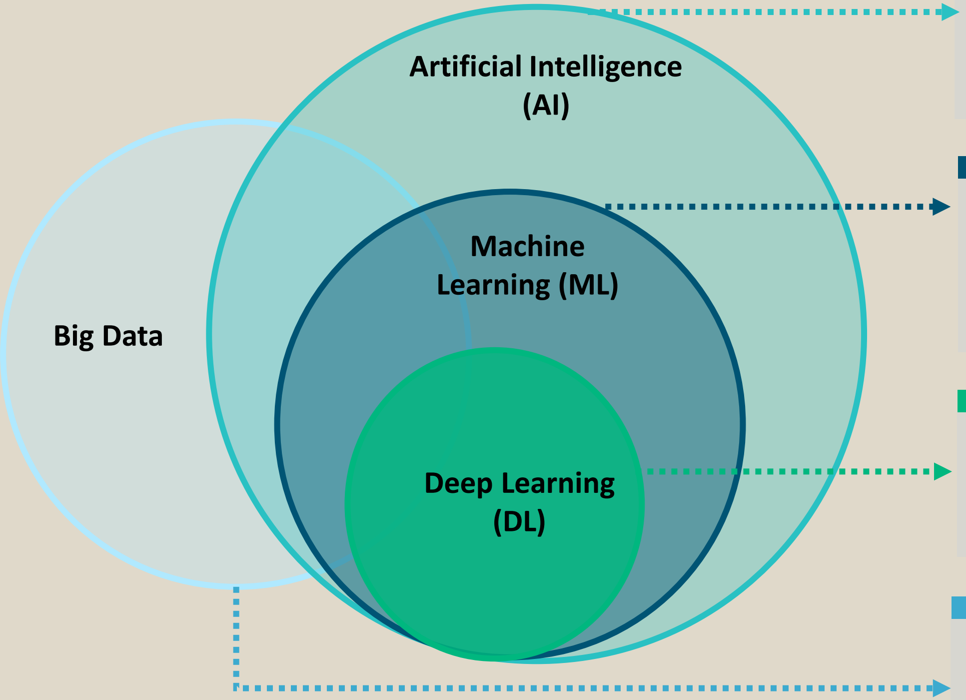
Module 7

Deep Learning and Image Recognition

Agenda

What is deep learning?

- Deep learning is a **specific subfield** of **machine learning**
- It is a new take on learning representations from data that puts an emphasis on **learning successive layers** of increasingly meaningful representations.
- These layers are built on **artificial neural networks** with representation learning.
- Learning can be **supervised**, **semi-supervised** or **unsupervised**.



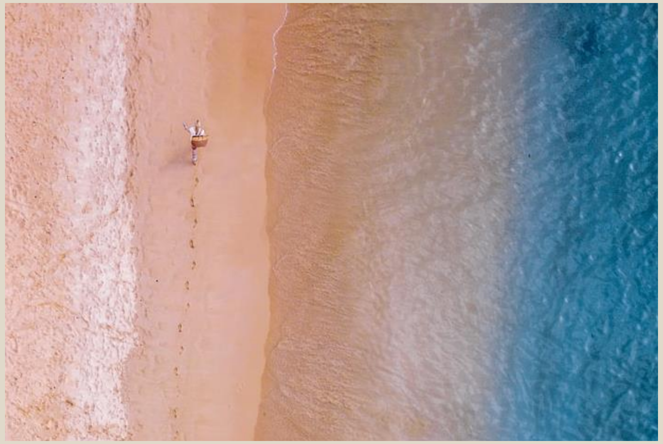
Artificial Intelligence – is the ability of computer systems to perform tasks that normally require human intelligence

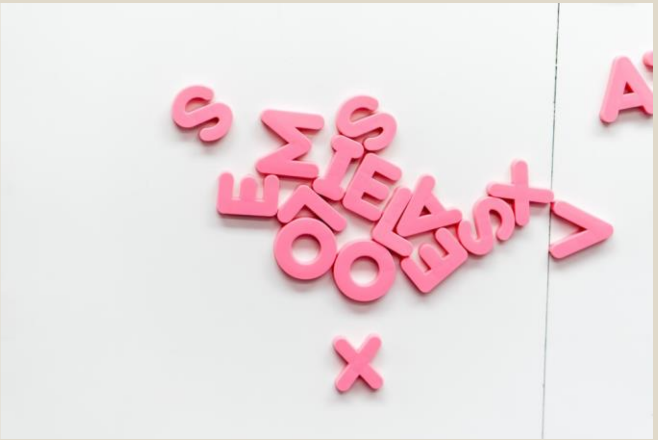
Machine Learning – is the capacity of a computer to learn from experience, i.e. to modify its processing on the basis of newly acquired information

Deep Learning – is a specific approach used for building and training Neural Networks

Big Data – Large, complex and often unstructured data

Applications and achievements of deep learning





More mature and industrial applications

In active development and research

Near-human-level image classification Improved machine learning	Near-human-level speech recognition Near-human-level handwriting transcription	Autonomous driving

Is this a car in the image?

1

- Have a database of cars and match the images by pixel value
  - Just one pixel changes and the match doesn't happen

→ Use database as training data

2

- Create a histogram over the colors
  - But cars can have different colors

→ Include the color as feature

3

- Build a car filter, e.g. for tires
  - How do we connect the different filters?

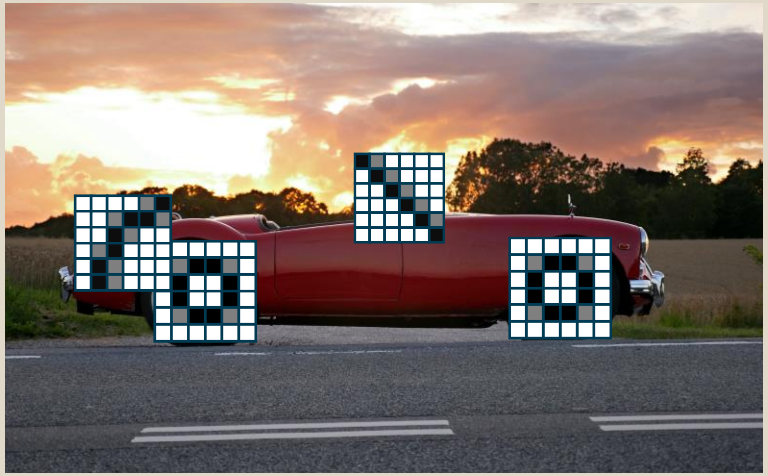
→ Detect shapes with features

4

- Use a machine learning model to combine the filter results

→ Combine the results with a trained SVM

Applying filters



Move the filter over image as a sliding window

3	1	1	2	8
1	0	7	3	2
2	3	5	1	1
1	4	1	2	6
3	2	1	3	7

Part of the image

Original image: 5x5

1	0	-1
1	0	-1
1	0	-1

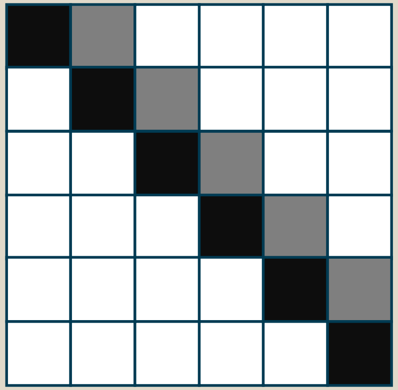
Filter

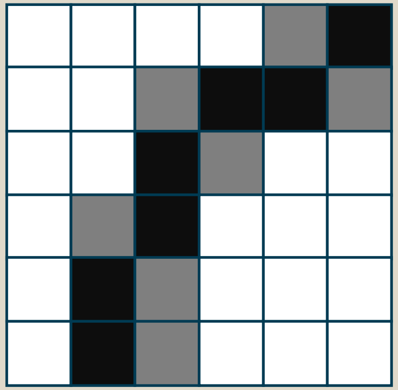
Filter 3x3

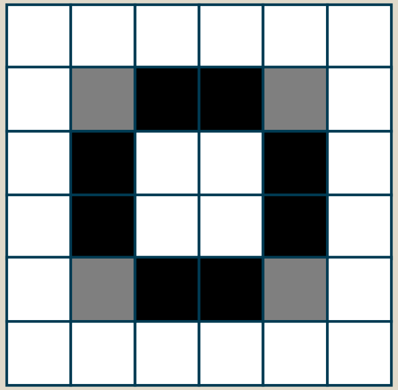
-7	...		
..	...		

Part of the image

Filtered image: 4x4

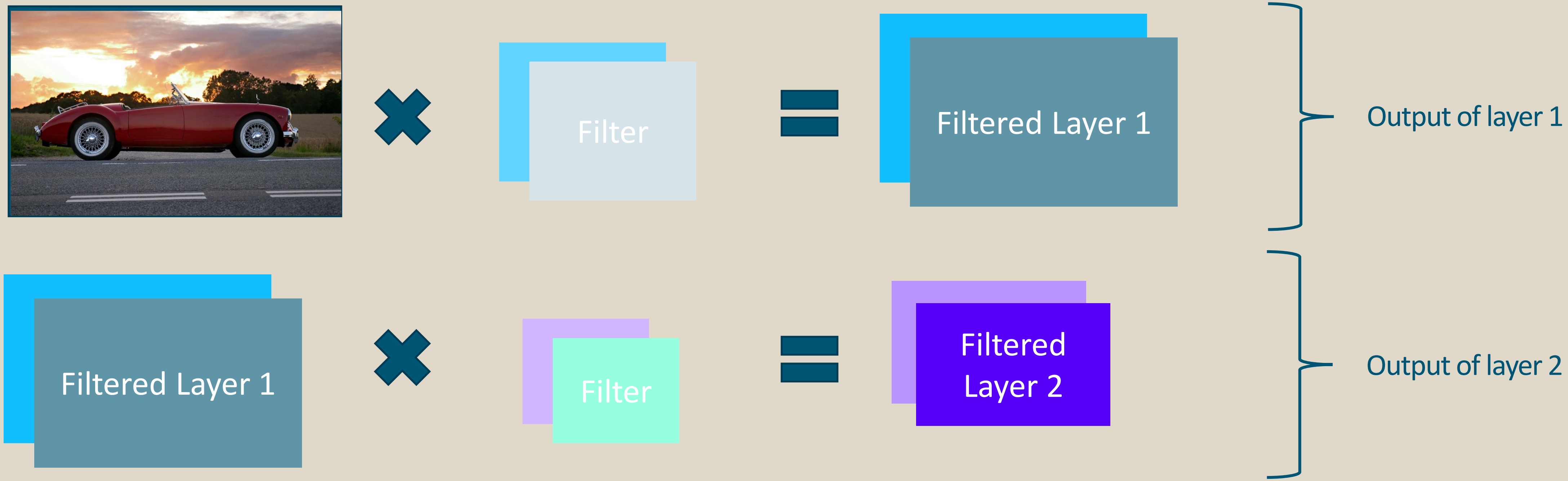






Convolutional layers and weight sharing

- A “small” image with 64x64 pixels and a neuronal network with a hidden layer size of 2000 neurons results in 8.2 million weights that need to be trained, for the first layer alone
- Instead use the filter approach from above. The filter weights are learned by backpropagation.
    - Which automates the feature extraction and engineering process
- Even if we use multiple filters per layer this reduces the number of weights drastically
    - E.g.: Using 64 3x3 filter results in 576 weights, that need to be trained
- Stacking multiple layers of filters allows the extraction of higher order features



Controlling the filtered size: Padding and stride

- For each layer the number of dimensions increases while size of the filtered image is decreasing.
- We can control the filtered image size the kernel size, the padding and stride
- The output image size can be calculated with: $n_{out} = \left[\frac{N_{in}+2p-k}{s}\right] + 1$
- Where n_{out} is the output size, $n_{in}$the image input size, p the padding, s the stride and k the filter size

Stride = 2

Padding = 1

	3	1	1	2	8	
	1	0	7	3	2	
	2	3	5	1	1	
	1	4	1	2	6	
	3	2	1	3	7	

Original image: 5x5

1	0	-1
1	0	-1
1	0	-1

Filter 3x3

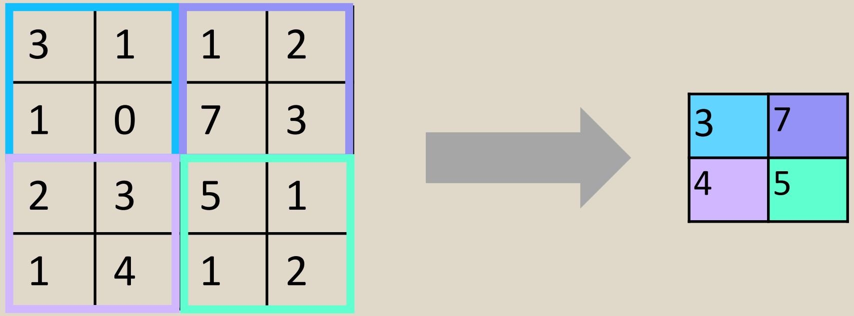
-?	…	
..	…	

Filtered image: 3x3

$\left[\frac{5+2-3}{2}\right] + 1 = 3$

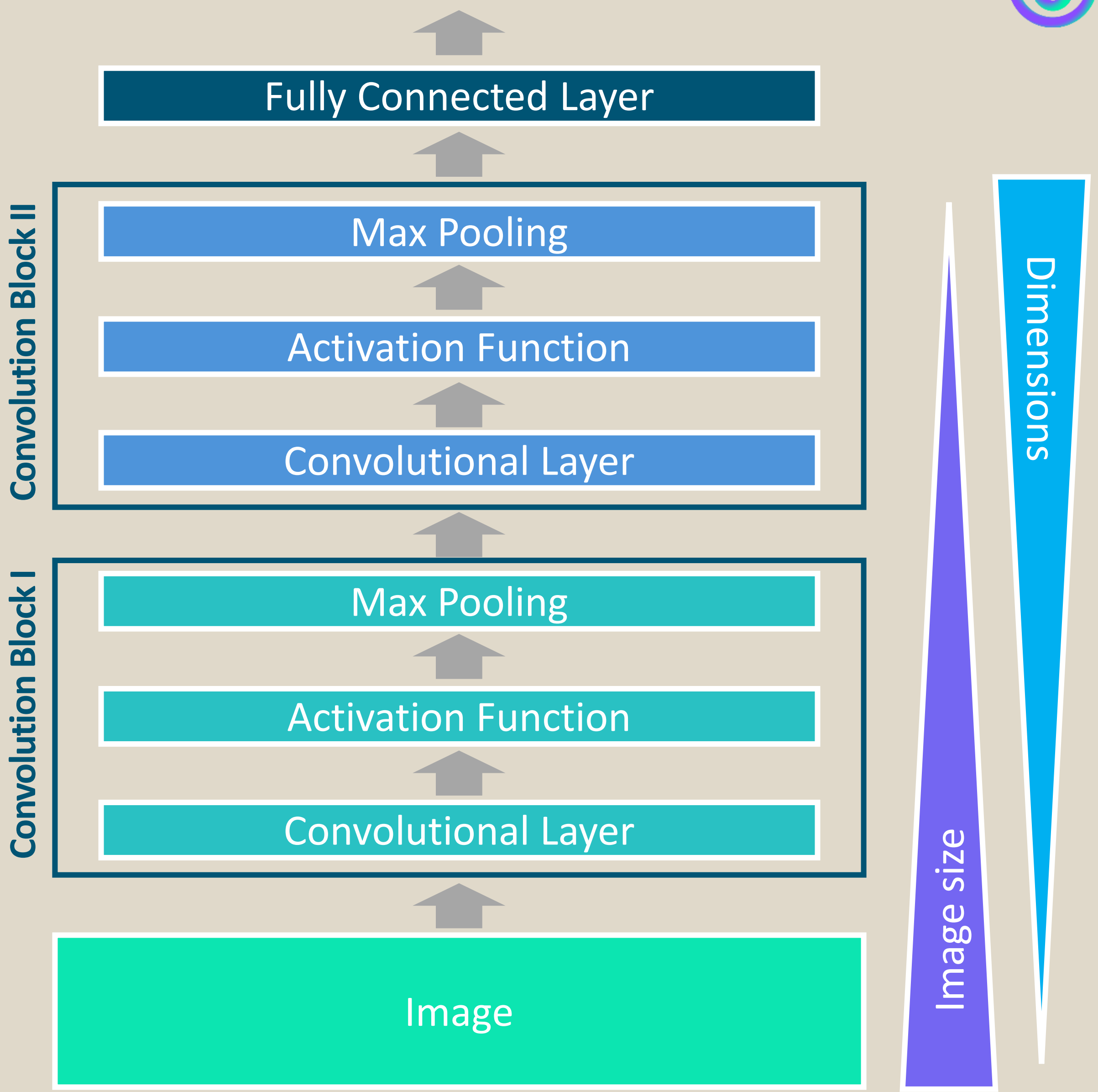
Reducing the filtered size: Max-pooling

- Using sliding window over the image results in overlapping filter results.
- The best matching filter will have the highest total result.
- We get the best matching result for a region by using another filter with stride=filter size that returns the maximum value



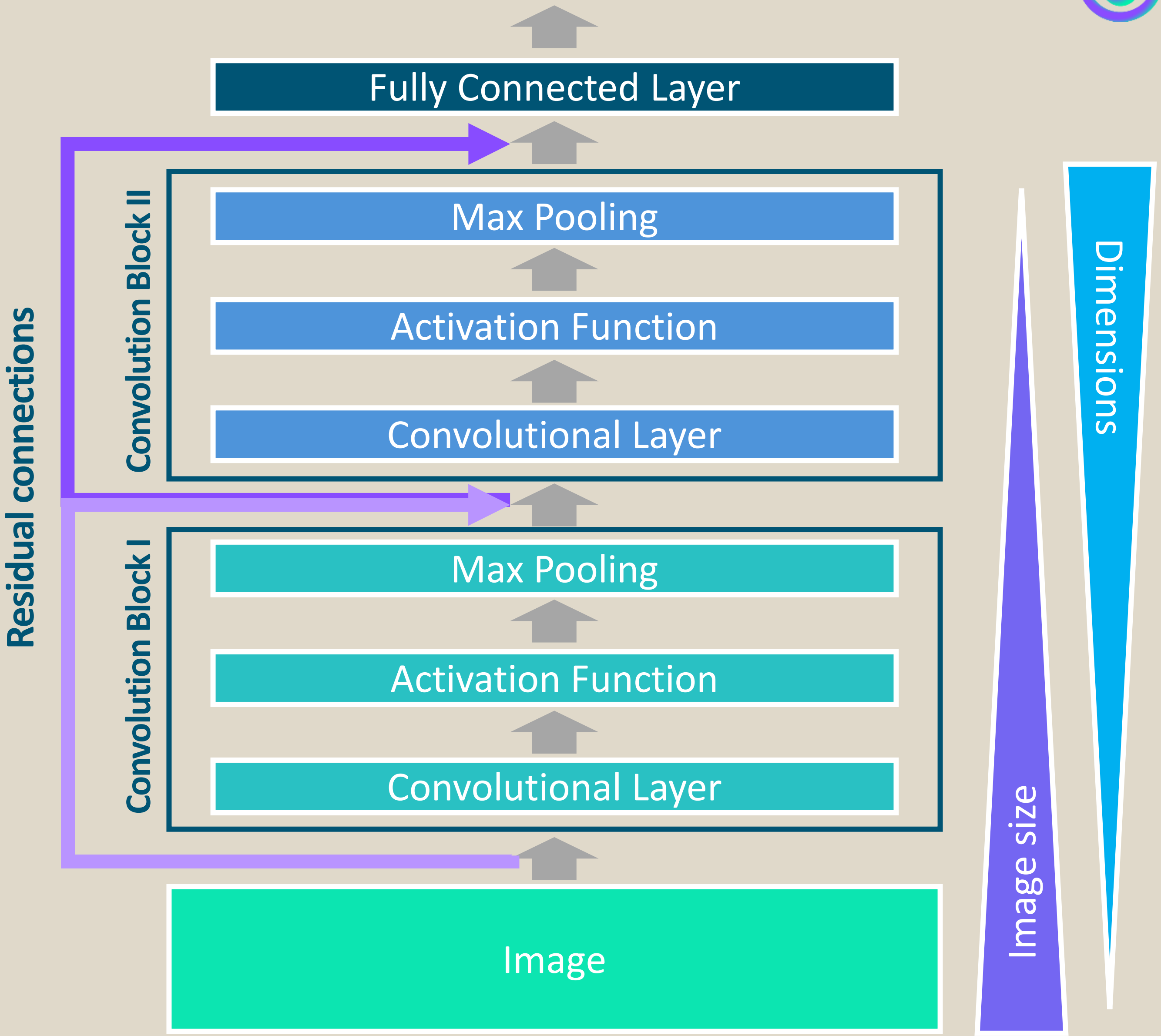
Convolutional Neural Networks

- Deep Neural Network are often built from repeated blocks
- This is done by repeatedly stacking convolution layers and max pooling layers
- The final layers are often fully connected and combine the filter results for the final classification
- In Computer vision the number of dimensions is often increased while the image size is reduced
- Pioneered by AlexNet



Increasing the depth: ResNet

- Increasing the depth of models by stacking more blocks had diminishing returns, because despite the changes from before the vanishing gradient is still a problem
- That’s why the residual connections help transport the gradient unhindered through the network
- We can now stack hundreds of layers
- Pioneered by Res Net (Residual Neural Network)



Recap: Training Neural Networks

Input data

- Training data used to train a deep learning model

Layers

- Each layer represents a data-processing module that takes input as one or more tensors and outputs to one or more tensors.

Weights

- Each layer is assigned a weight that represents its relative importance, which is calculated based on its stochastic gradient descent.

Predictions

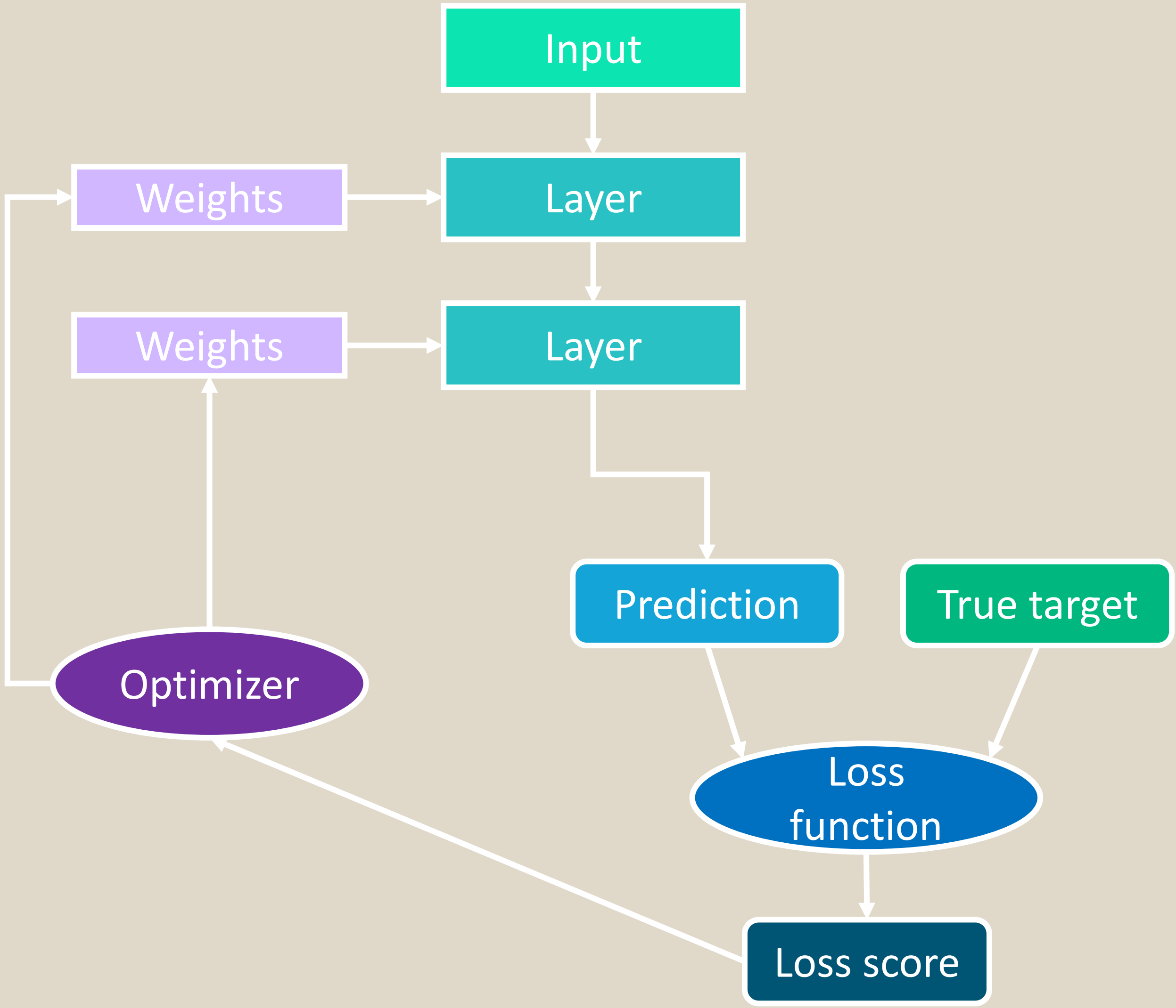
- Predicted outputs which are normally compared against true targets for performance evaluation

Loss function and loss score

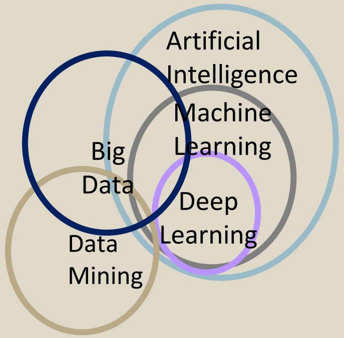
- A measure of how successful the model is at hand. Loss score is used to determine the quantity to be adjusted during training to improve performance.

Optimizer

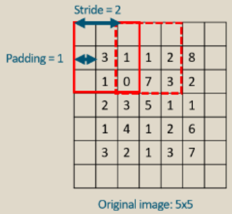
- Determines how the network will be updated based on the loss function. It is calculated based on stochastic gradient descent (SGD).



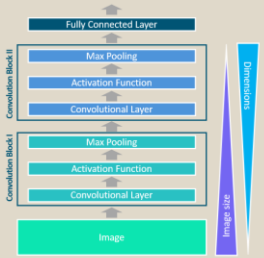
Important takeaways



Deep Learning is a subfield of Artificial Intelligence and is used for Problem with large data, like
- Image recognition
- Natural language processing



Filter layers are used as edge detectors. By combining the output of the edge detectors, a prediction about the content of the image is made. Parameters controlling the filter process are **filter size, padding and stride**.



Convolutional Neural Networks (CNN) automate the feature engineering steps normally involved in image recognition. A CNN consists of layers of **Convolutional** and **Max-pooling layers**.

Try it yourself!

In the following exercises

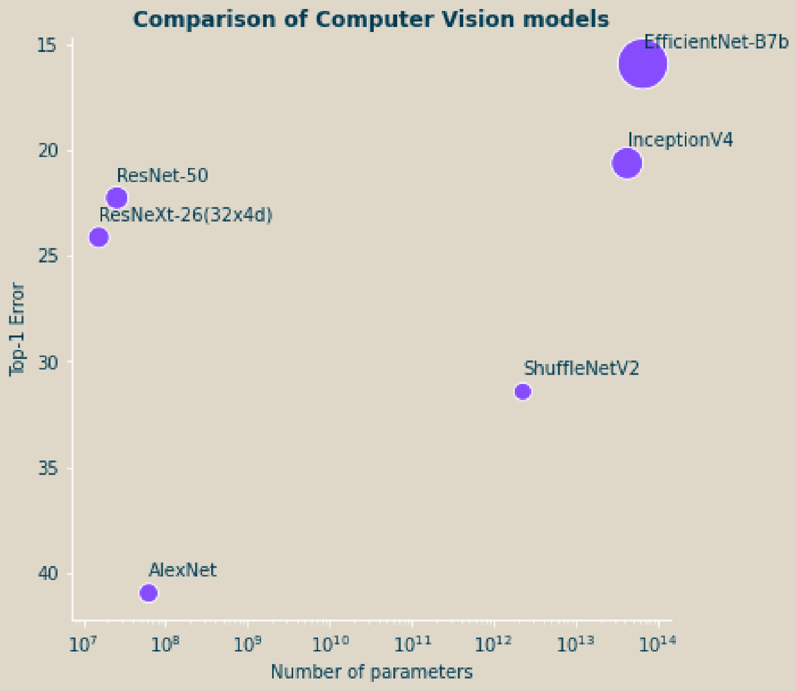
Transfer-learning

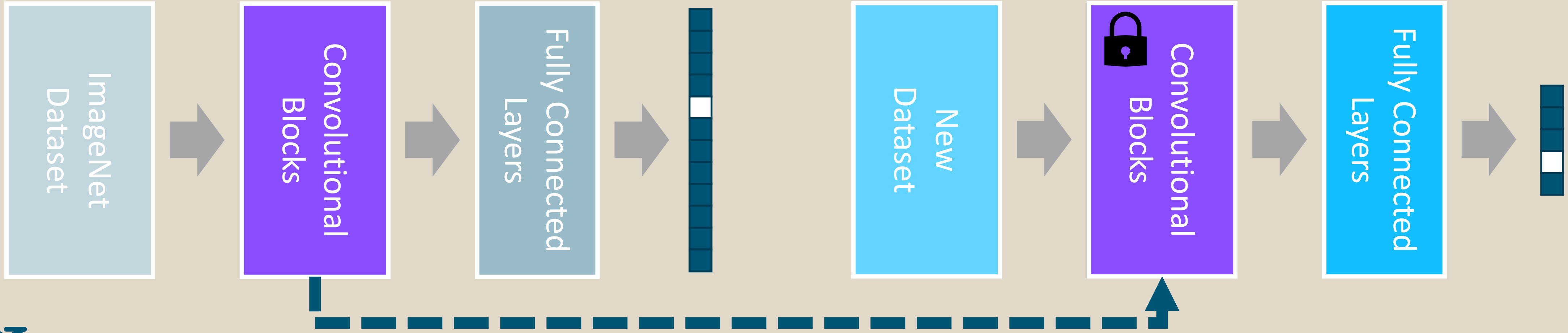
Training a ResNet-50 takes 14 days

- The ResNet-50 has 25.5M parameters (one of the smaller models)
- The dataset is ImageNet
- The training is done on a single GPU

Training image recognition models yourself is not feasible

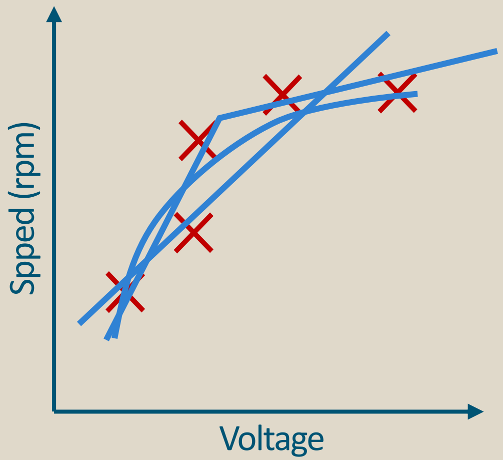
- Instead, we want to reuse the learned model



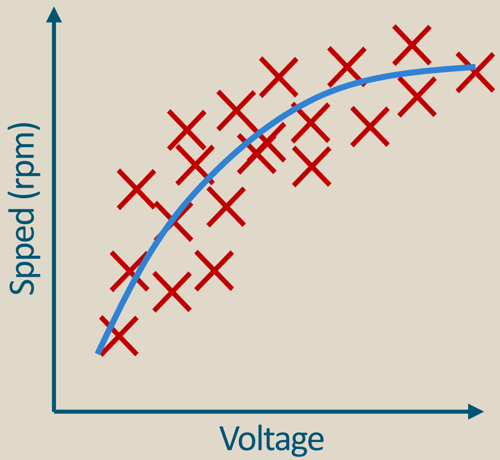


Data-centric AI

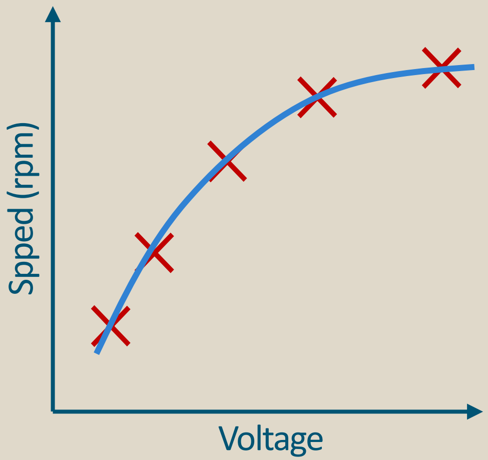
Model-centric view	• Collect what data you can. • Hold the data fixed and iteratively improve the code/model.
Data-centric view	• Use tools to improve the data quality. • Hold the code fixed and iteratively improve the data.



- Small data
- Noisy labels



- Big data
- Noisy labels



- Small data
- Clean (consistent) labels

Data-centric: Having high quality data leads to significant improvements in performance

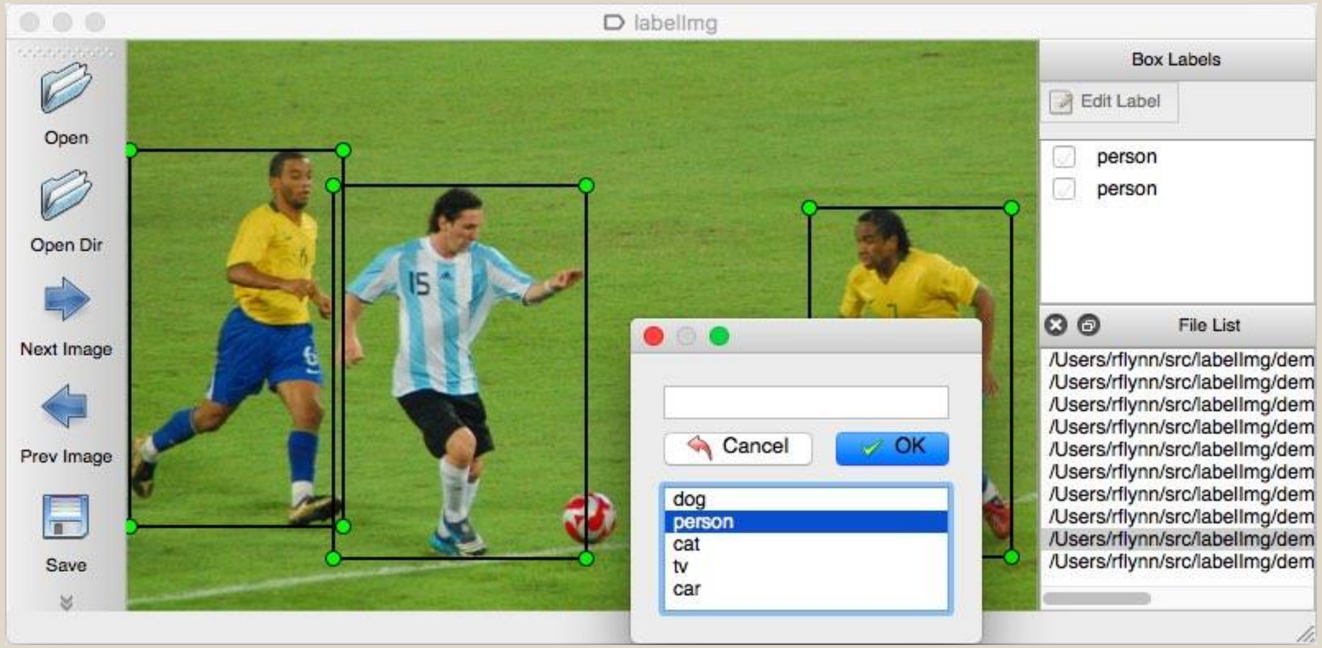
Labeling for transfer-learning

For transfer learning, we need only a few high quality samples

We can use platforms and tools like “LabelImg“ or Amazon Mechanical Turk for the labeling process

Data-centric approach: Iteratively improve the data

- Train a model
- Error analysis to identify the types of data the algorithm does poorly on
- Either get more of that data via data augmentation, data generation or data collection
- Make sure to cover different situations in lighting, backgrounds, angles and reflections
- Give more consistent definition for labels if they were found to be ambiguous



Tzutalin. LabelImg. Git code (2015). https://github.com/tzutalin/labelImg

CNN for object detection: (Faster) R-CNN

R-CNN: Region-based convolutional neural net

The model consists of three parts

- Backbone network which can be a pre-trained CNN
- Region proposal network (RPN) which finds and proposes regions of interest (Regions that contain an object)
- Classification network which performs a classification on each region of interest

For the R-CNN all models are combined into one Neural Network

- The overall loss is the sum of the regression and the classification

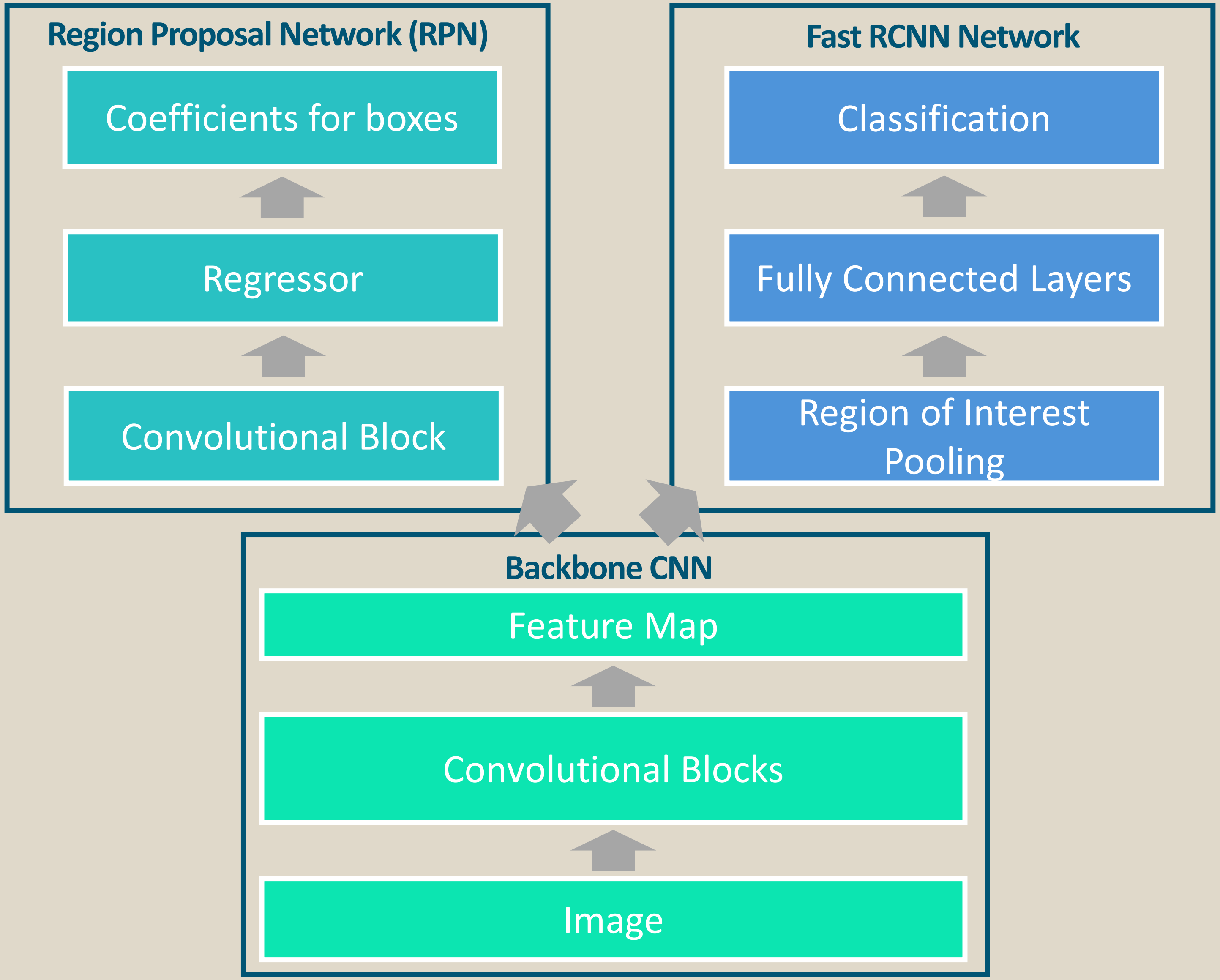


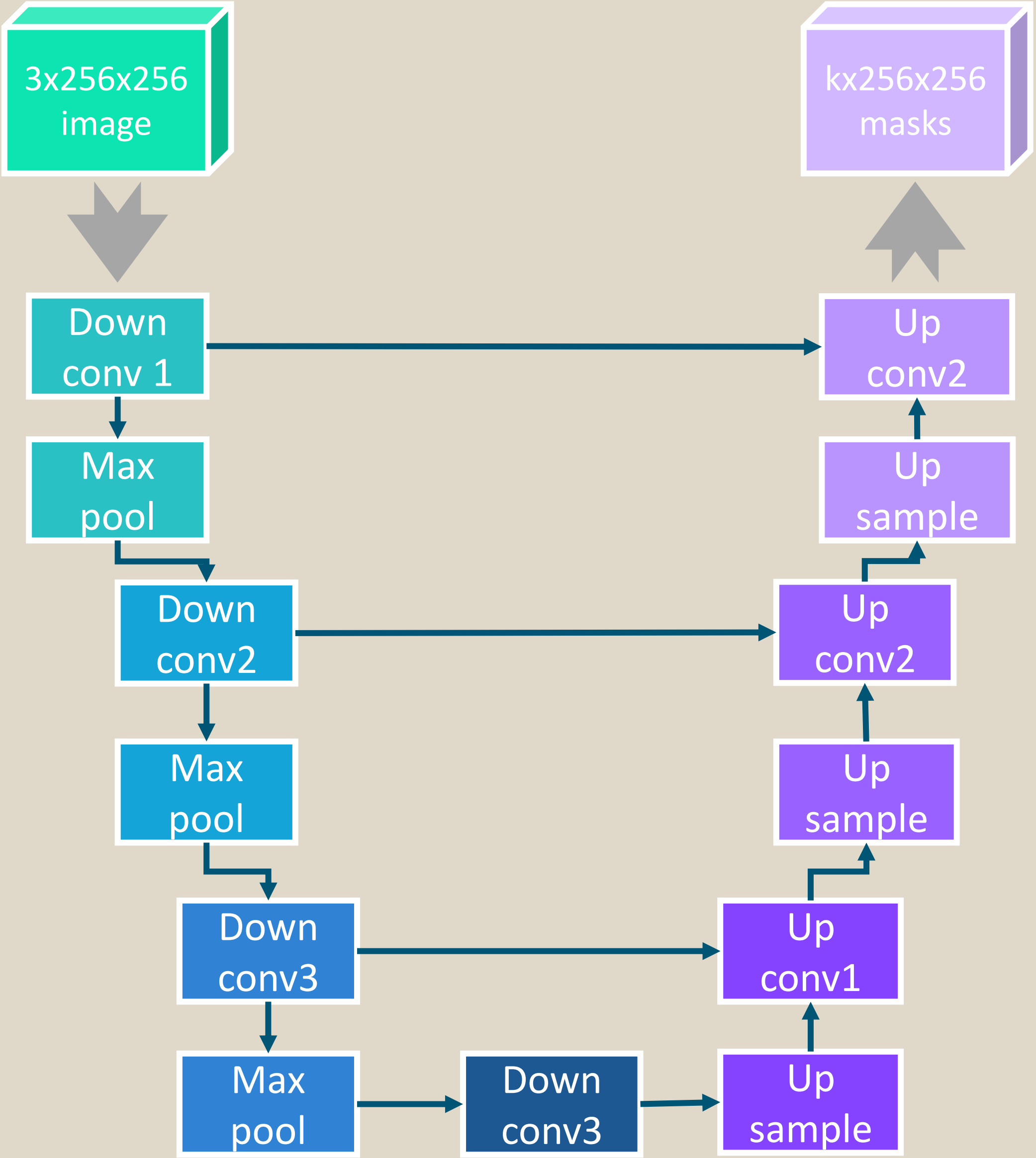
Image segmentation: U-Net

Developed for biomedical image segmentation

- Data efficient and accurate

Algorithm

- With each max-pooling operation the image gets downsampled
  - Forces the model to generalize and reduce information
- The information is than upsampled again (reconstructs information)
  - The reason for the name U-Net
- Use skip-connection to reduce information loss and retain context information



Face Recognition with one-shot learning

Face Detection/Tracking

- Locate one or more faces in the image and mark with a bounding box

Face Alignment

- Normalize the face to be consistent with the database, such as geometry and photometrics

Feature Extraction

- Extract features from the face that can be used for the recognition task

Face Recognition

- Perform matching of the face against one or more known faces in a prepared database

Face 1 Person A | Face 2 Person A | Face 3 Person B

Deep Learning model

Convolutional Blocks

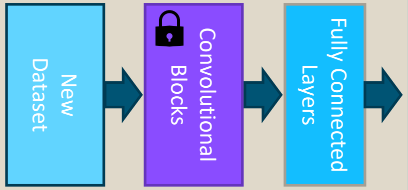
Fully Connected Blocks

Facial Features

Loss

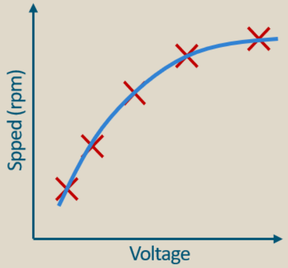
- Optimize the weights so that the features for face 1 and 2 should be maximally similar while the distance between face 1&3 and face 2&3 should be maximally different

Important takeaways



We can reuse state-of-the-art image recognition models by retraining them on a different dataset.

- Retraining reduces the data required
- Retraining reduces the compute power required



With a reduced number of training samples it is of high importance to choose the samples carefully and correct any inconsistencies in the labeling process. This is called **data-centric view**.

For many advanced use cases in image recognition, we can use pre-trained filters. The advanced cases of this session were:

- R-CNN
- U-Net
- Face Recognition

In most practical cases we reuse state-of-the-art models and architectures and make only small changes to it

Try it yourself!

In the following exercises

Agenda

Feedback and Q&A

Thank you

If you would like any further information please contact Werner,Dr.,Fabian_Georg (BI X) BIX-DE-I <fabian_georg.werner@boehringer-ingelheim.com>