ML Workshop

Eine Einführung in

Machine Learning & Natural Language Processing

Aufbau & Konzept

Theorie zum Verständnis

Praktische Übungen zur Festigung

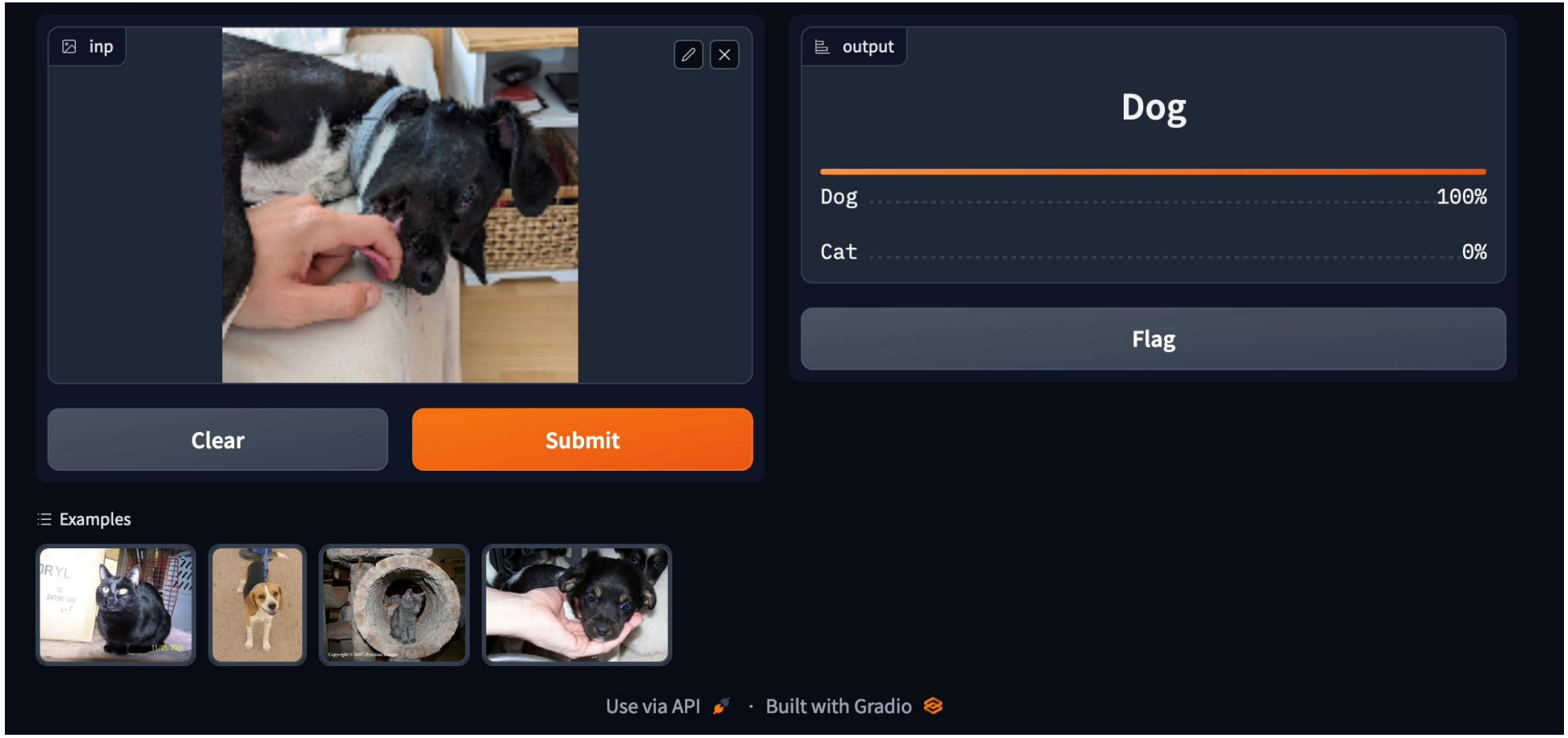
Interaktive Ergebnisse zum Experimentieren

Tag 1 – Grundlagen

Tag 1 - Grundlagen

- Überblick über die ML Landschaft
- Der ML Entwicklungsprozess
- Arbeiten mit Jupyter Notebooks
- Wie funktionieren Neuronale Netze?
- Was ist ein Convolutional Neural Network (CNN)?

Was bauen wir heute?



Eine Taxonomie der ML Landschaft

Einsatzzwecke – Eine Auswahl

Tabellarisch

- Tabular Classification
- Tabular Regression

Bilder

- Depth Estimation
- Image Classification
- Object Detection
- Image Segmentation
- Image-to-Image
- Unconditional Image Generation
- Video Classification
- Zero-Shot Image Classification

Text

- Text Classification
- Token Classification
- Question Answering
- Translation
- Summarization
- Conversational
- Text Generation
- Text2Text Generation
- Fill-Mask
- Sentence Similarity

Audio

- Text-to-Speech
- Automatic Speech Recognition
- Audio Classification
- Voice Activity Detection

Andere

- Feature Extraction
- Text-to-Image
- Image-to-Text
- Visual Question Answering

Arten von Learning

Supervised	Unsupervised
Semi-Supervised	Reinforcement Learning

Algorithmen

Klassisch

- Lineare Regression
- Decision Tree
- Random Forest
- k-NN
- k-Means Clustering
- Support Vector Machine
- Naive Bayes
- XGBoost
- Self-Organizing Maps

Deep Learning

Allgemein

- Feedforward
- Convolutional Neural Network (CNN)
- Transformer

Generativ

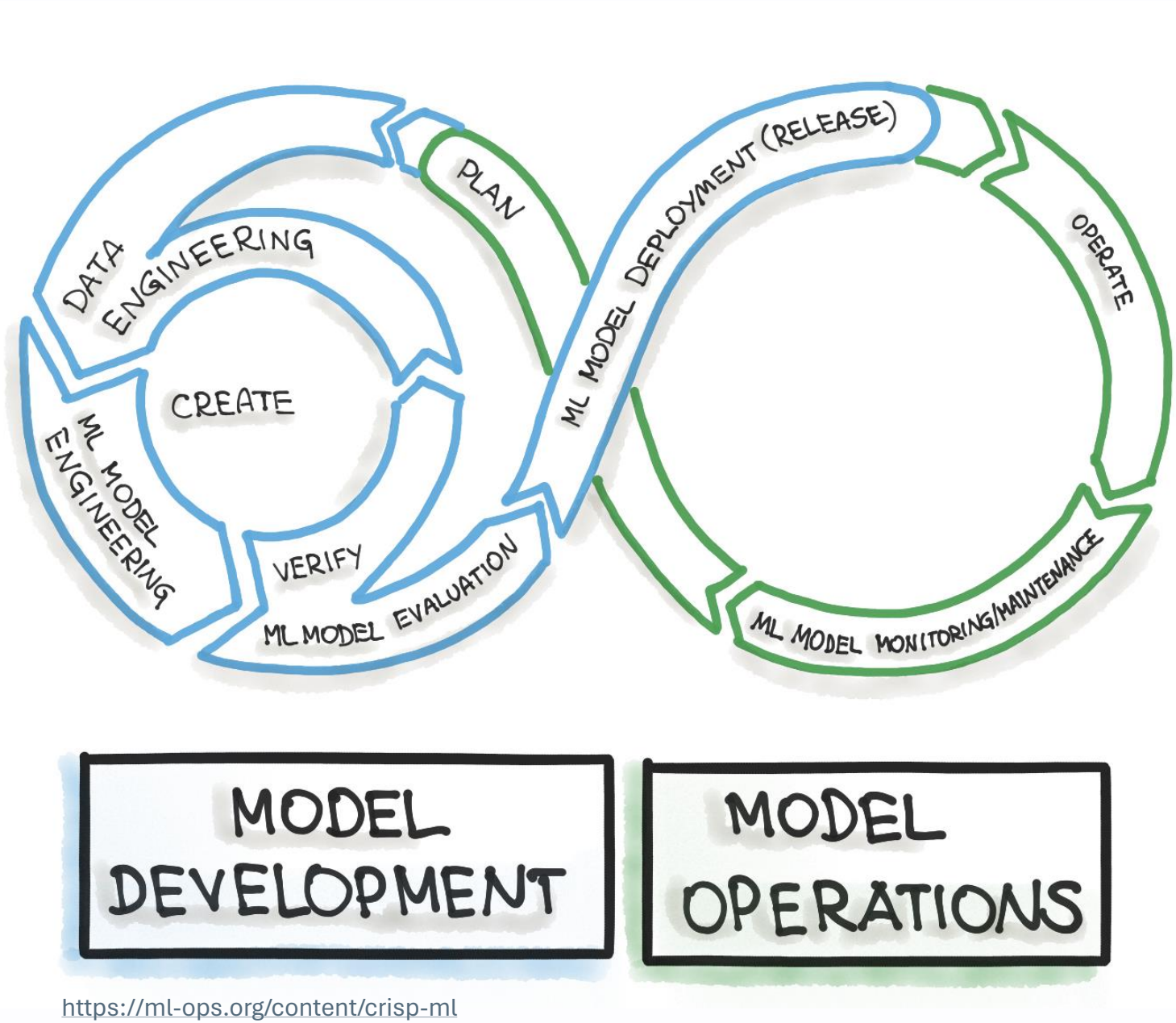
- Variational Autoencoder
- Generative Adversarial Network (GAN)

Auto-Regressiv

- LSTM
- GRU
- Transformer Decoder

Der Machine-Learning Entwicklungsprozess

Ablauf

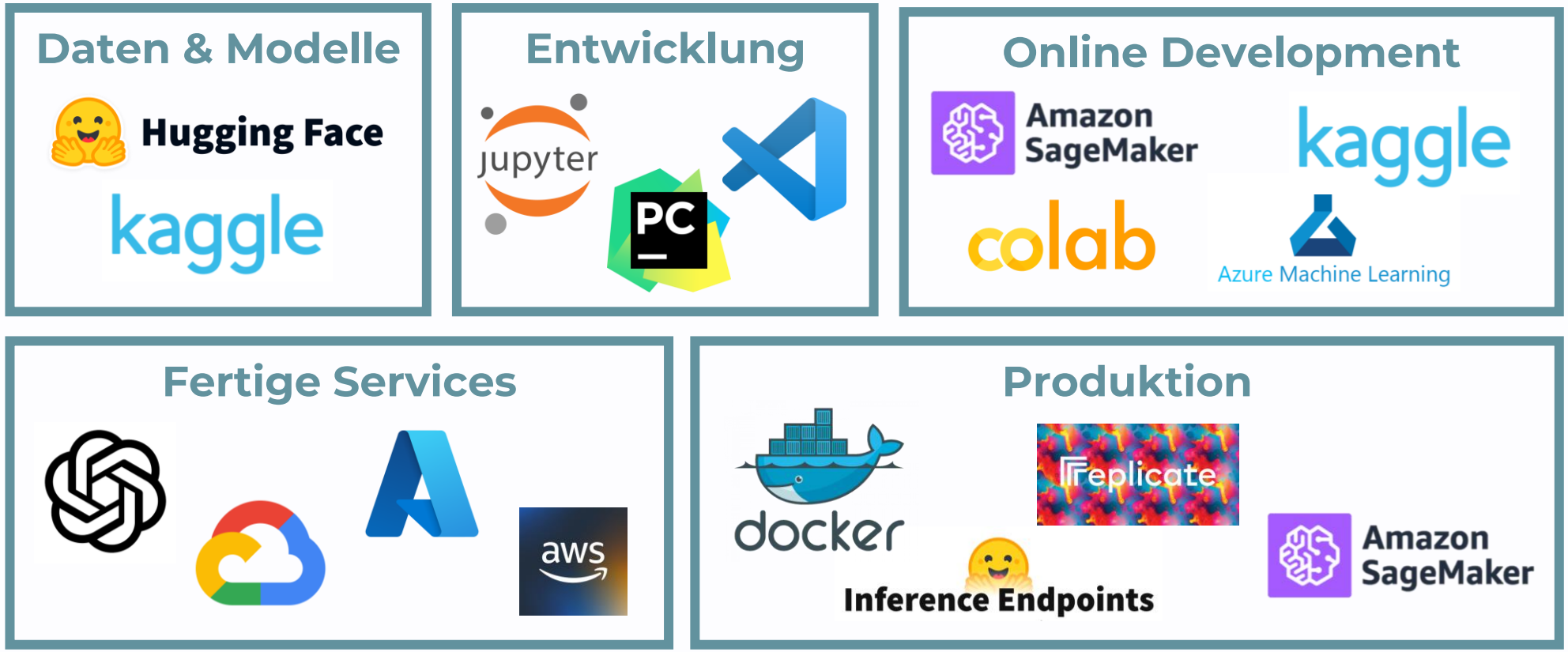


https://ml-ops.org/content/crisp-ml

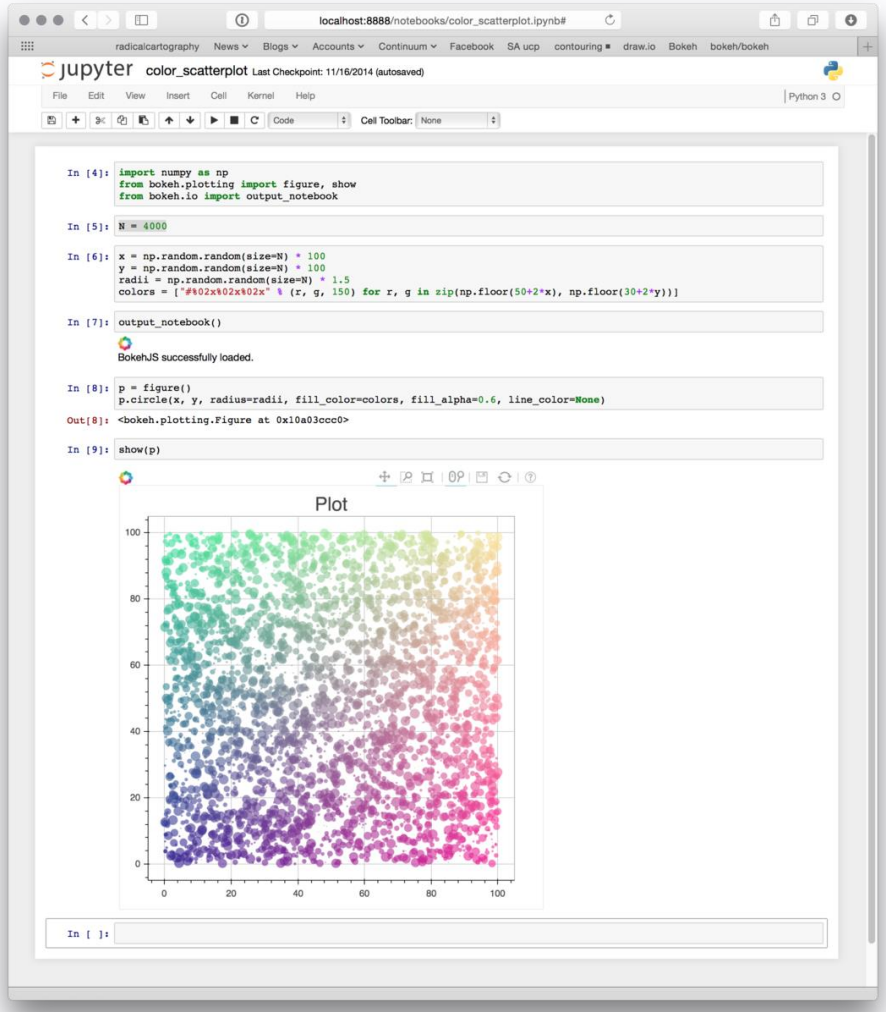
Frameworks

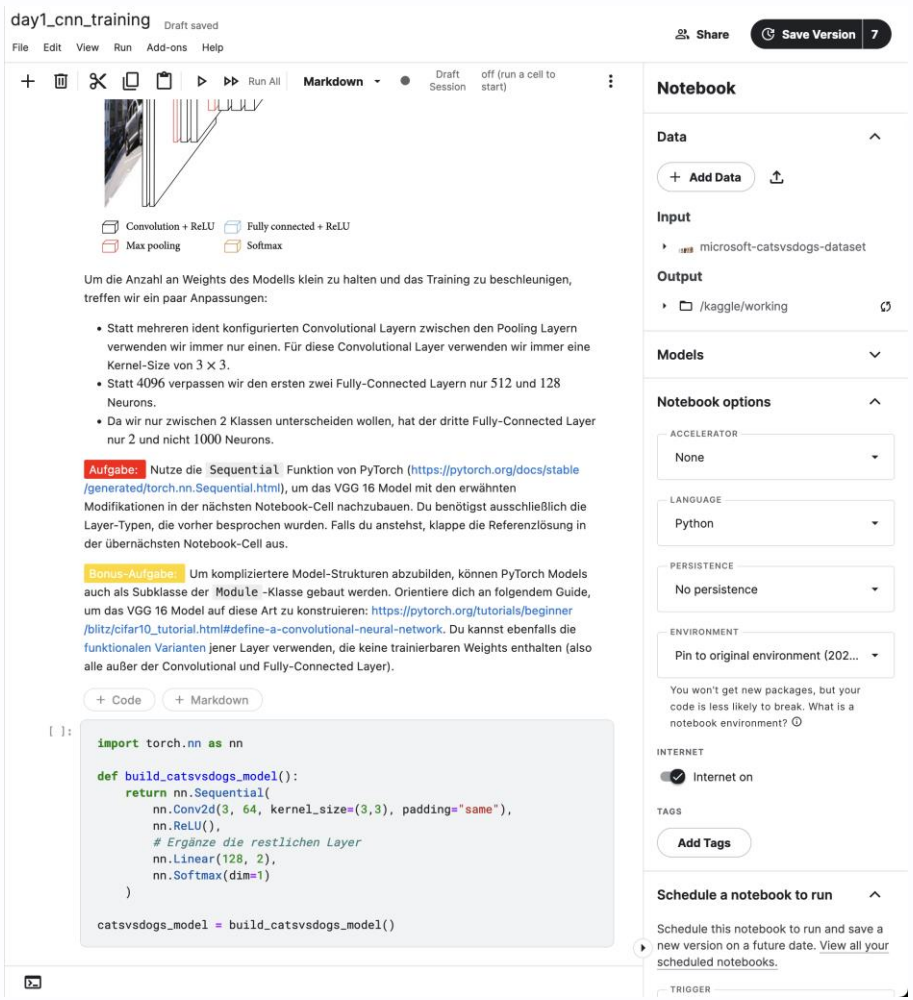

Sprachen
python
R
julia
Deep Learning
TensorFlow
PyTorch
Traditionelles ML
scikit learn
dmlc XGBoost
Inferenz
ONNX
TensorRT
Data Science
pandas
matplotlib
tidyverse
NumPy
APACHE Spark
gradio
Bilddaten
torchvision
OpenCV
pillow
Textdaten
LangChain
LLaMA C++
spaCy

Werkzeuge


Daten & Modelle
Hugging Face
kaggle
Entwicklung
jupyter
PC
Online Development
Amazon SageMaker
kaggle
colab
Azure Machine Learning
Fertige Services
aws
Produktion
docker
Replicate
Inference Endpoints
Amazon SageMaker

Jupyter Notebooks



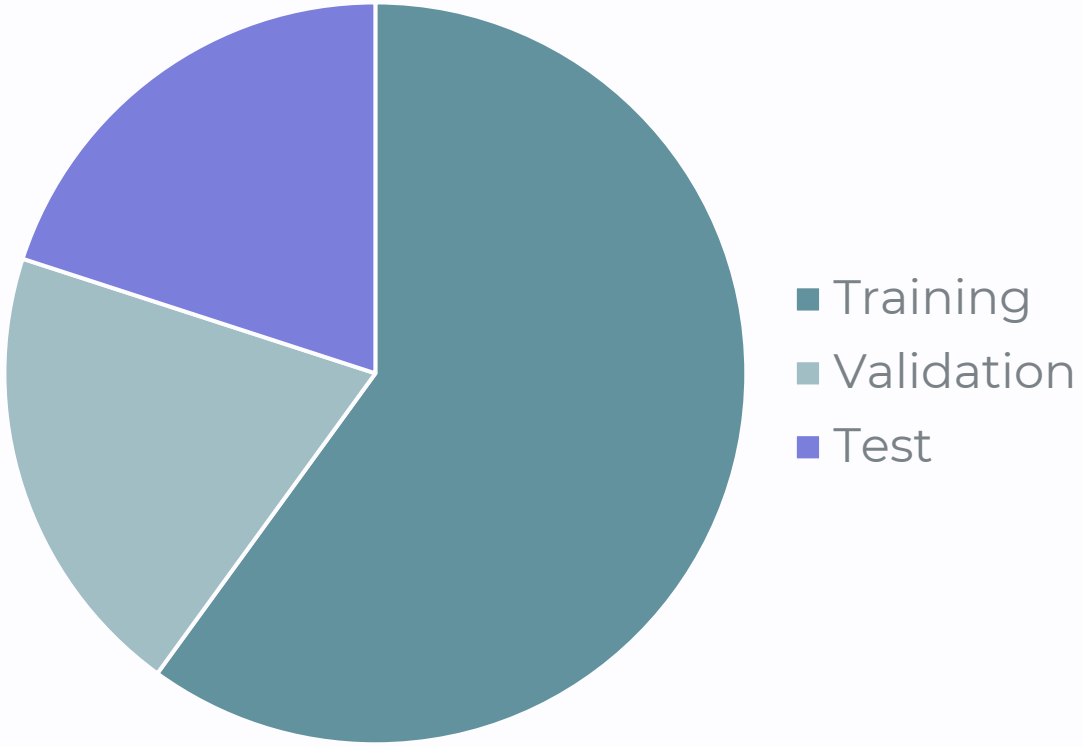


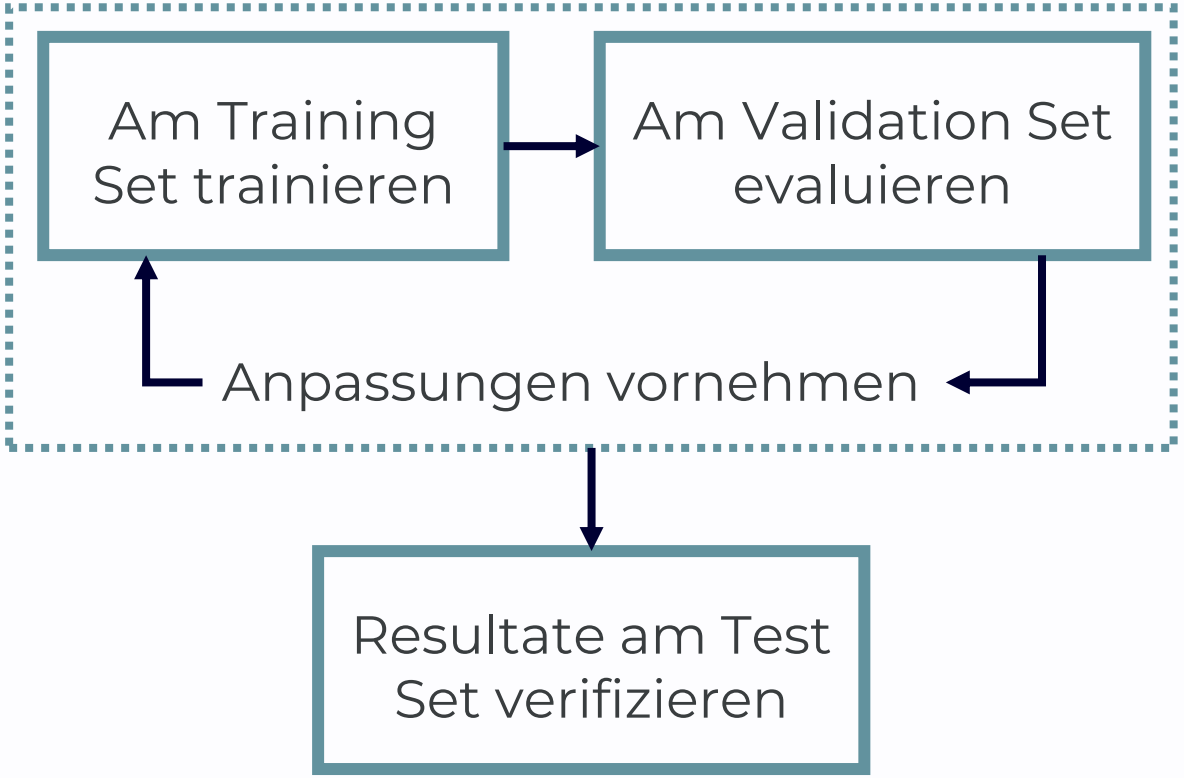
Übung 0

Jupyter Einführung & Kaggle Setup

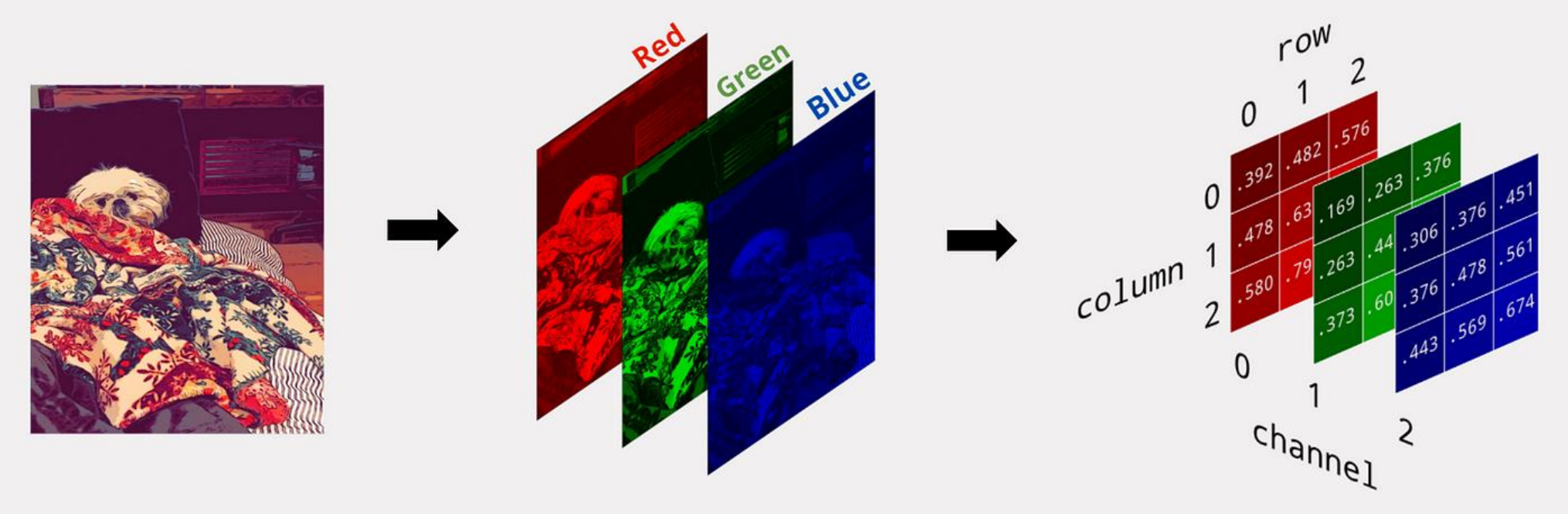
Daten

Daten-Aufteilung





Bilddaten als Tensor



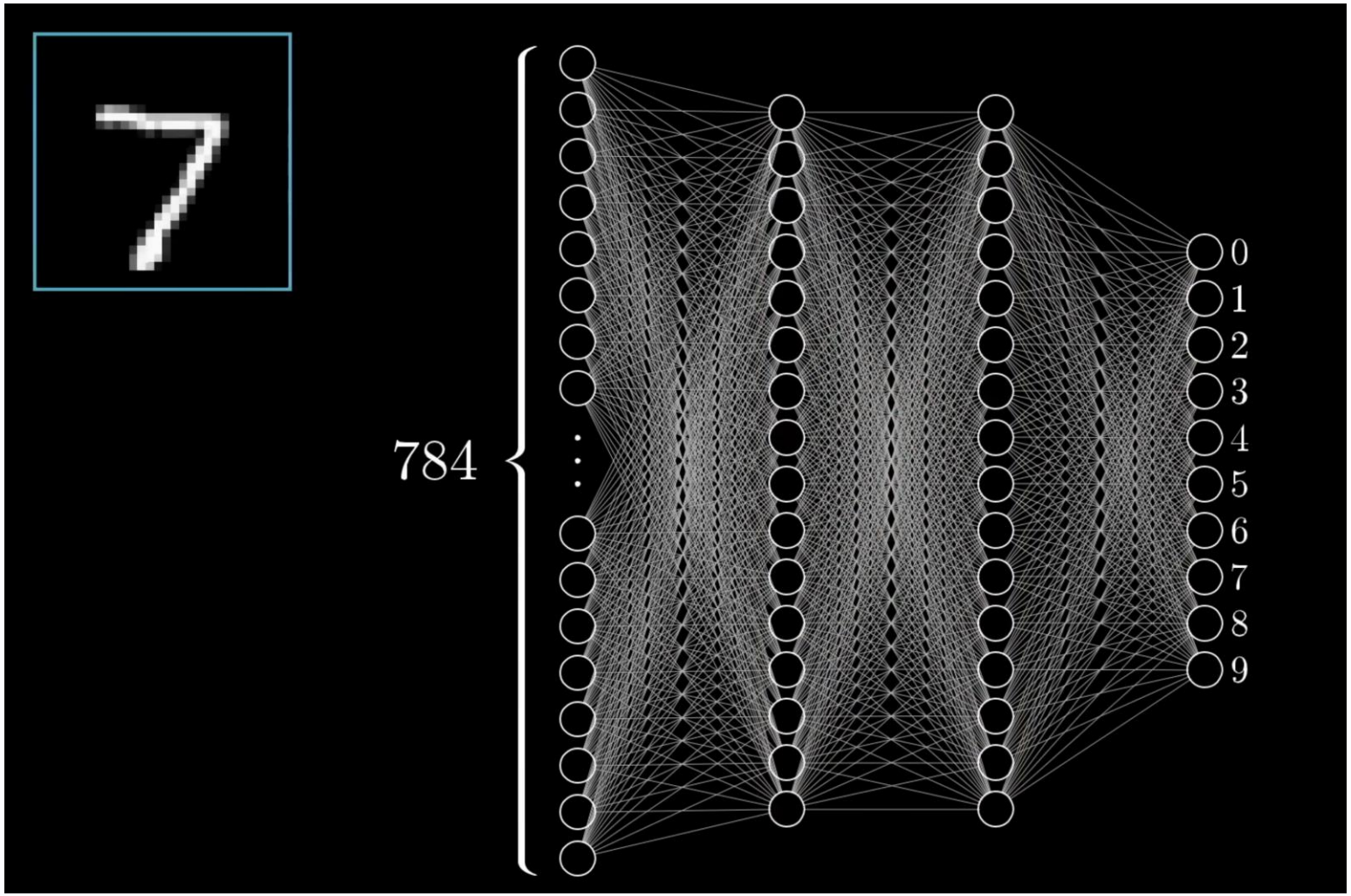
https://pub.towardsai.net/image-classification-with-python-cnn-vs-transformers-fe509cbbc2d0

Übung 1

Daten & Pre-Processing

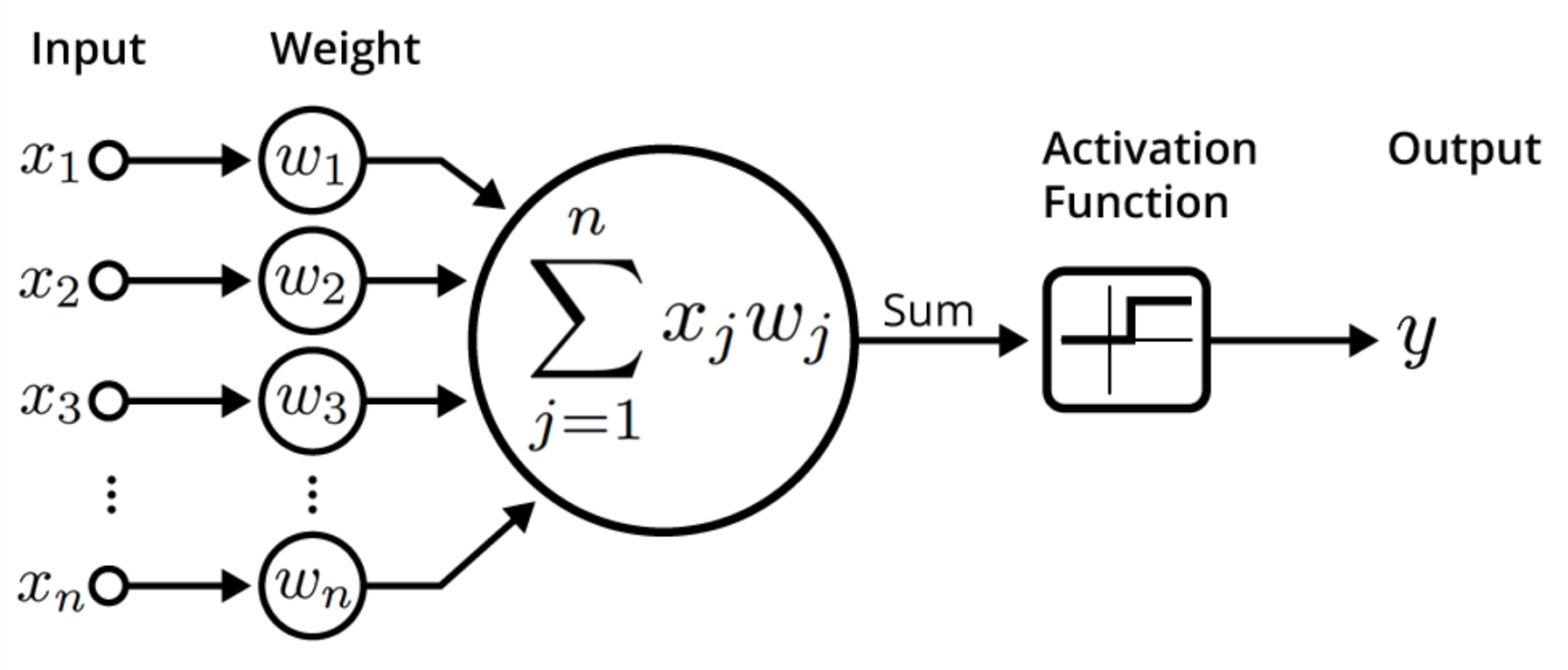
Wie funktioniert ein Neuronales Netz?

Die Grundstruktur



https://www.youtube.com/watch?v=aircAruvnKk

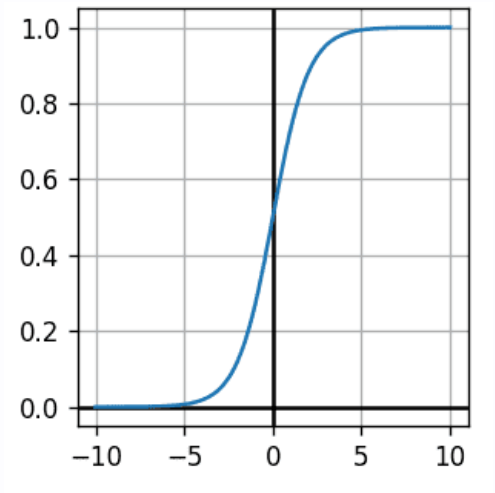
Ein Artificial Neuron



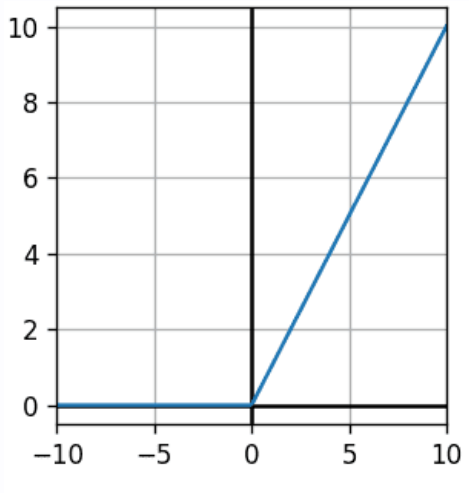
https://insights.sei.cmu.edu/blog/deep-learning-going-deeper-toward-meaningful-patterns-in-complex-data/

Activation Functions

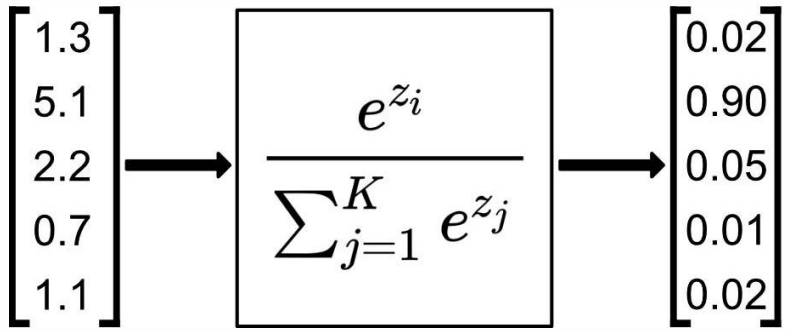
Sigmoid



ReLU



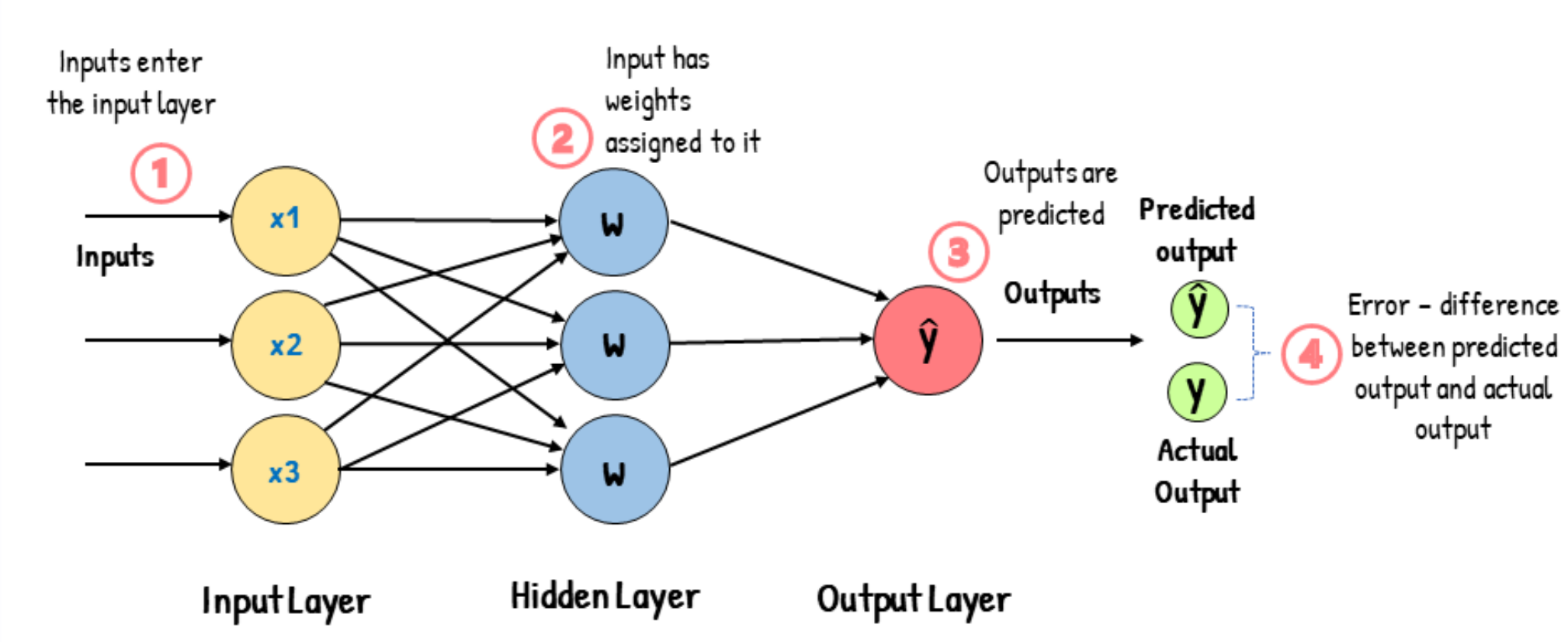
Softmax



https://dhruvs.space/posts/ml-basics-issue-4/

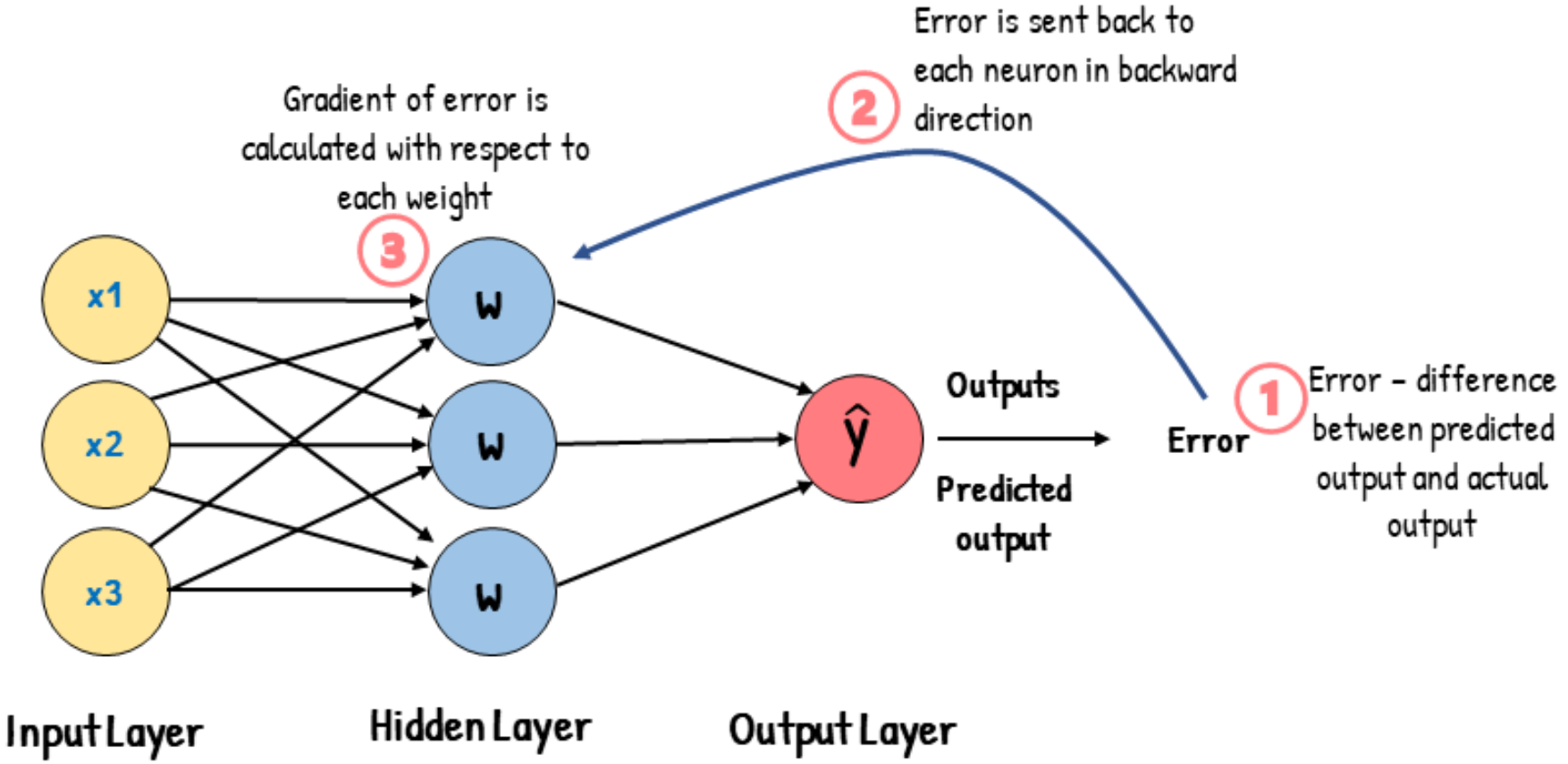
https://towardsdatascience.com/softmax-activation-function-explained-a7e1bc3ad60

Backpropagation 1/2



https://www.analyticsvidhya.com/blog/2023/01/gradient-descent-vs-backpropagation-whats-the-difference/

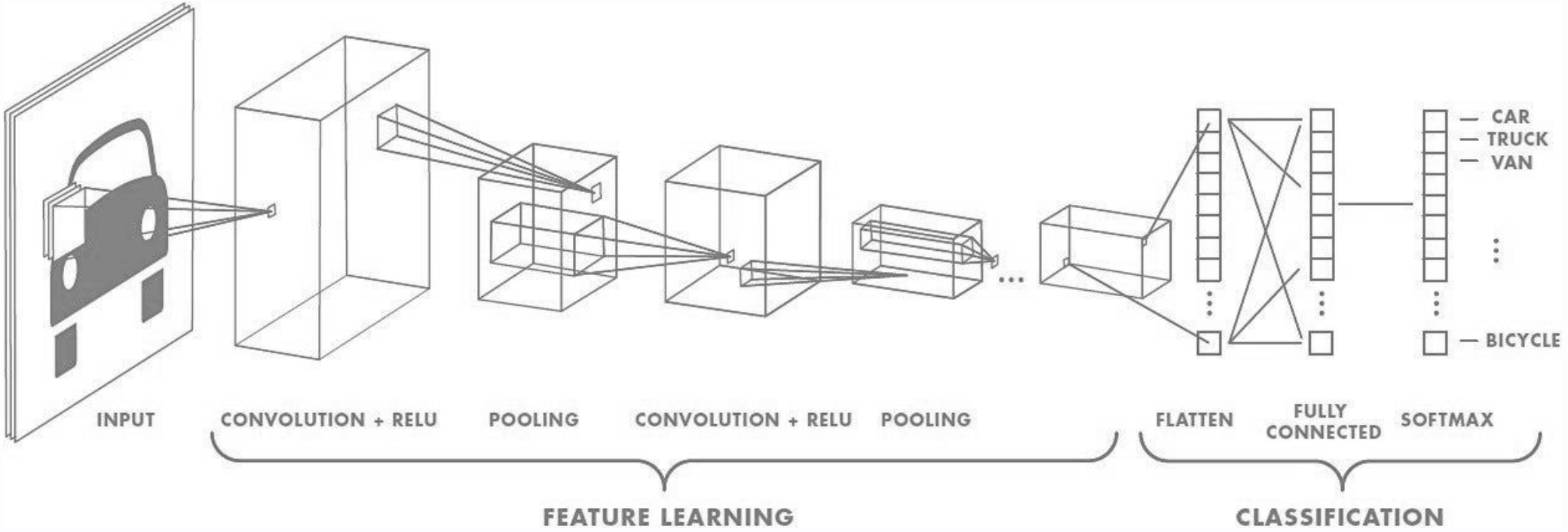
Backpropagation 2/2



https://www.analyticsvidhya.com/blog/2023/01/gradient-descent-vs-backpropagation-whats-the-difference/

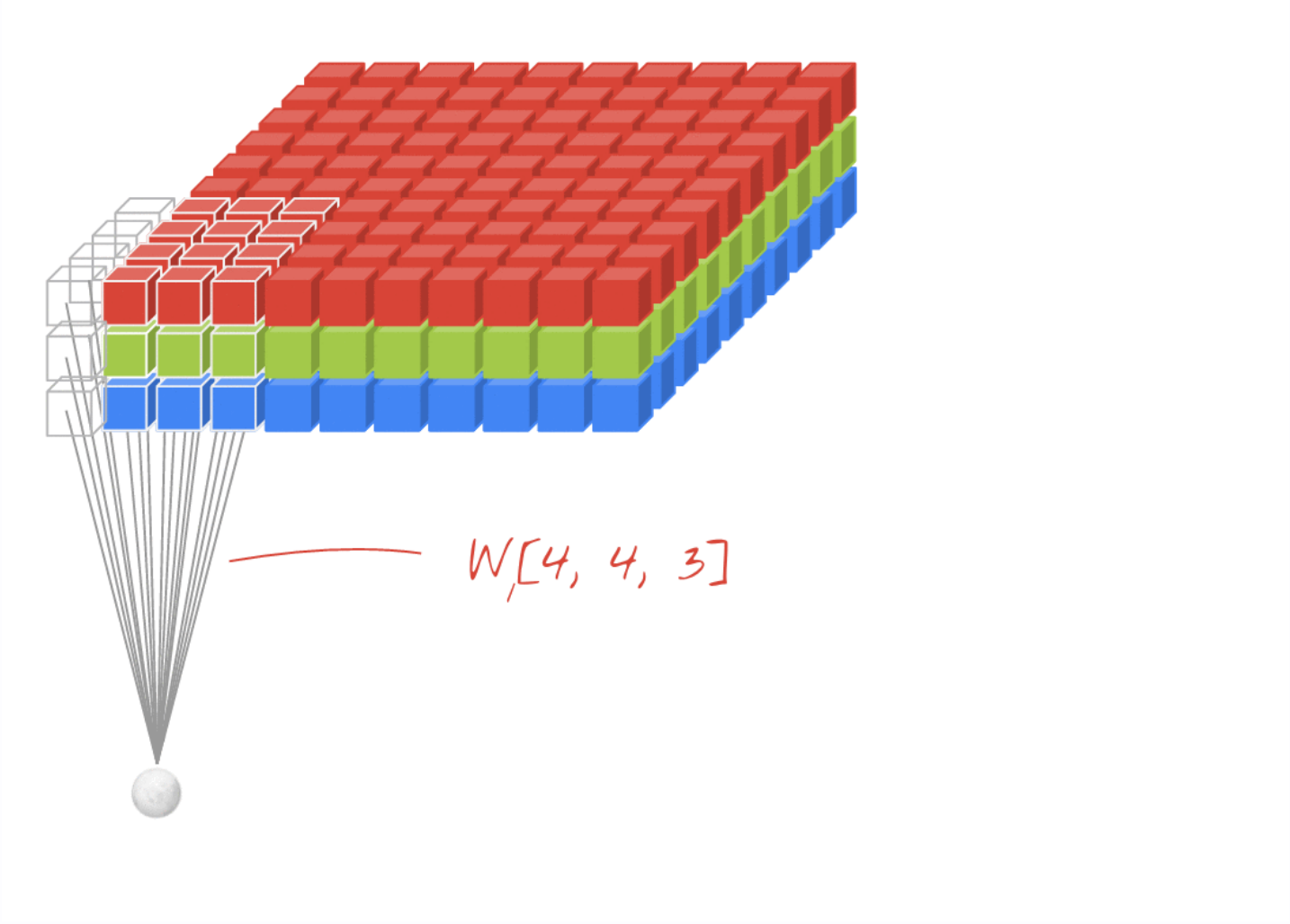
Convolutional Neural Networks

CNN Architektur



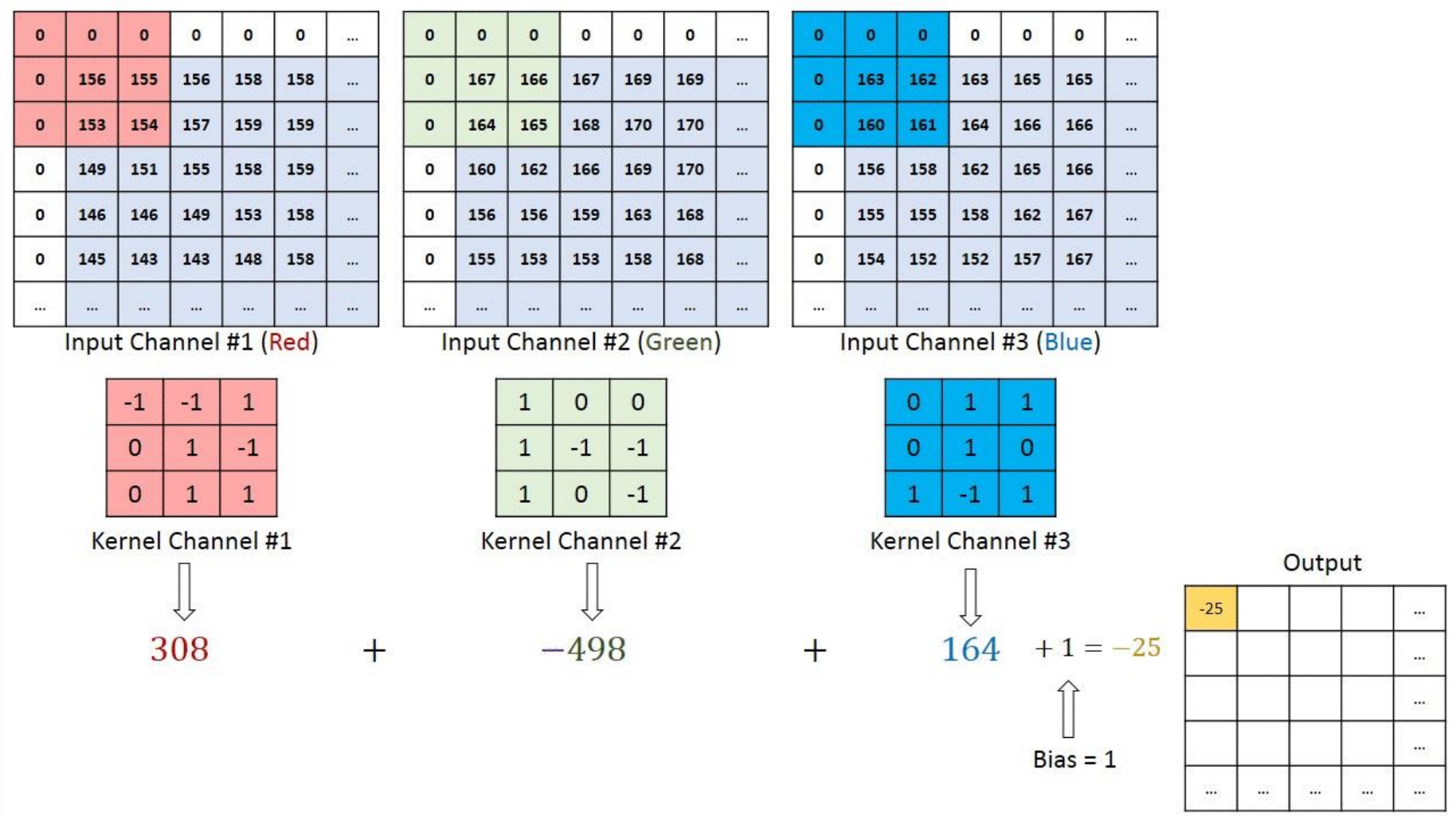
https://de.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html

Convolution 1/2



https://medium.com/latinxinai/vectorized-convolution-operation-using-numpy-b122fd52fba3

Convolution 2/2



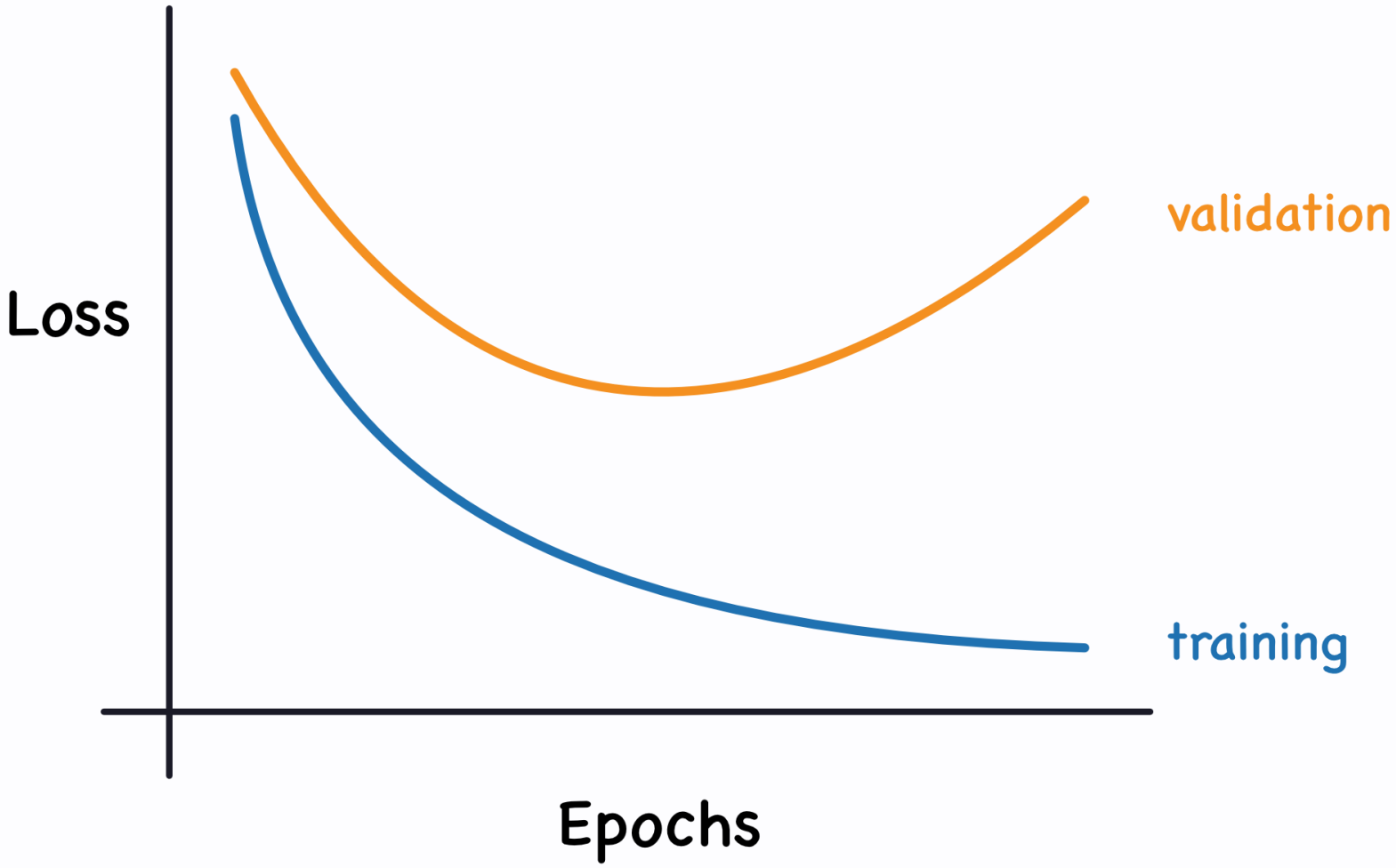
https://medium.datadriveninvestor.com/convolutional-neural-networks-3b241a5da51e

Übung 2

CNN Konstruktion & Training

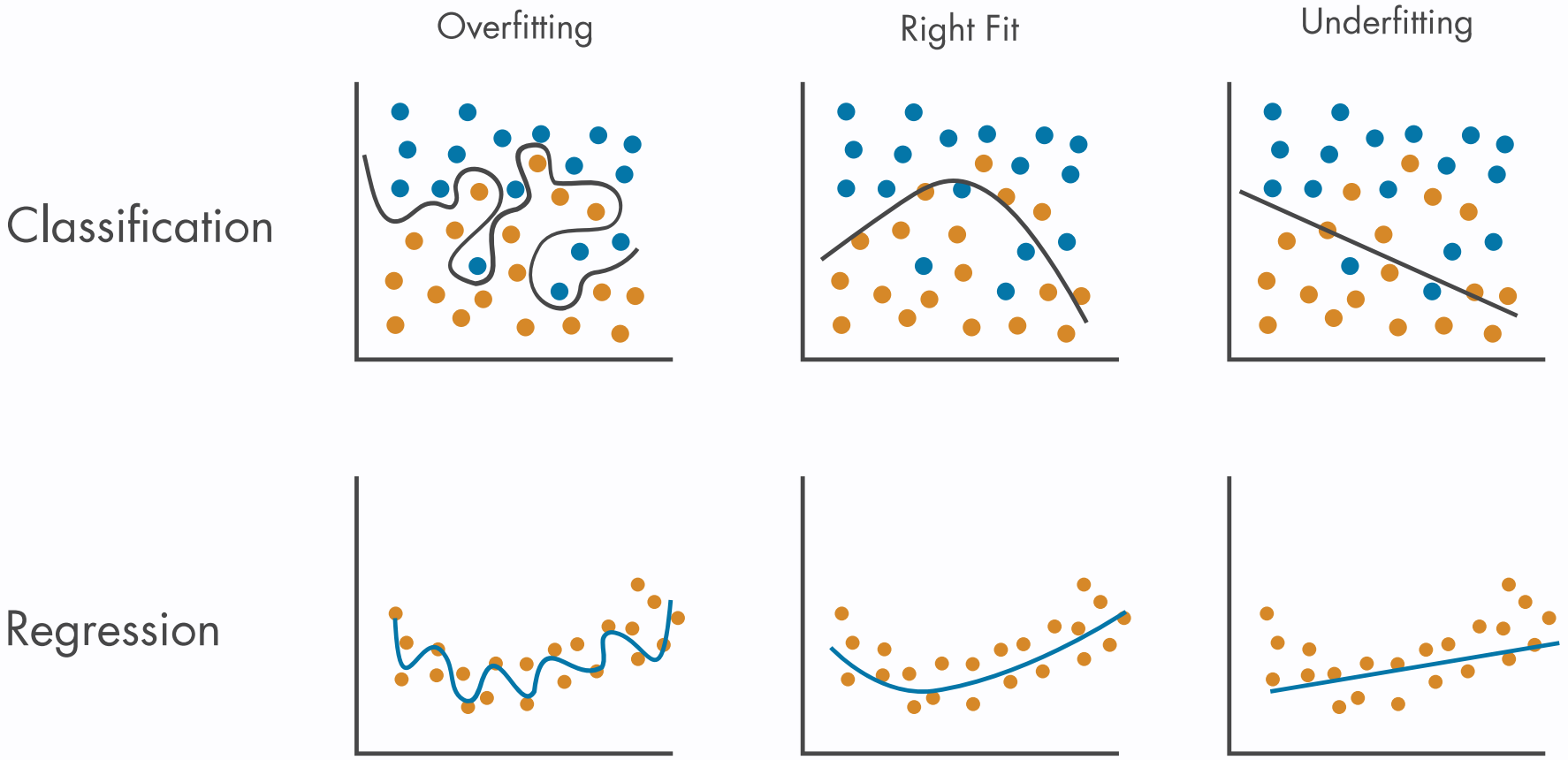
Overfitting

Training vs. Validation Loss



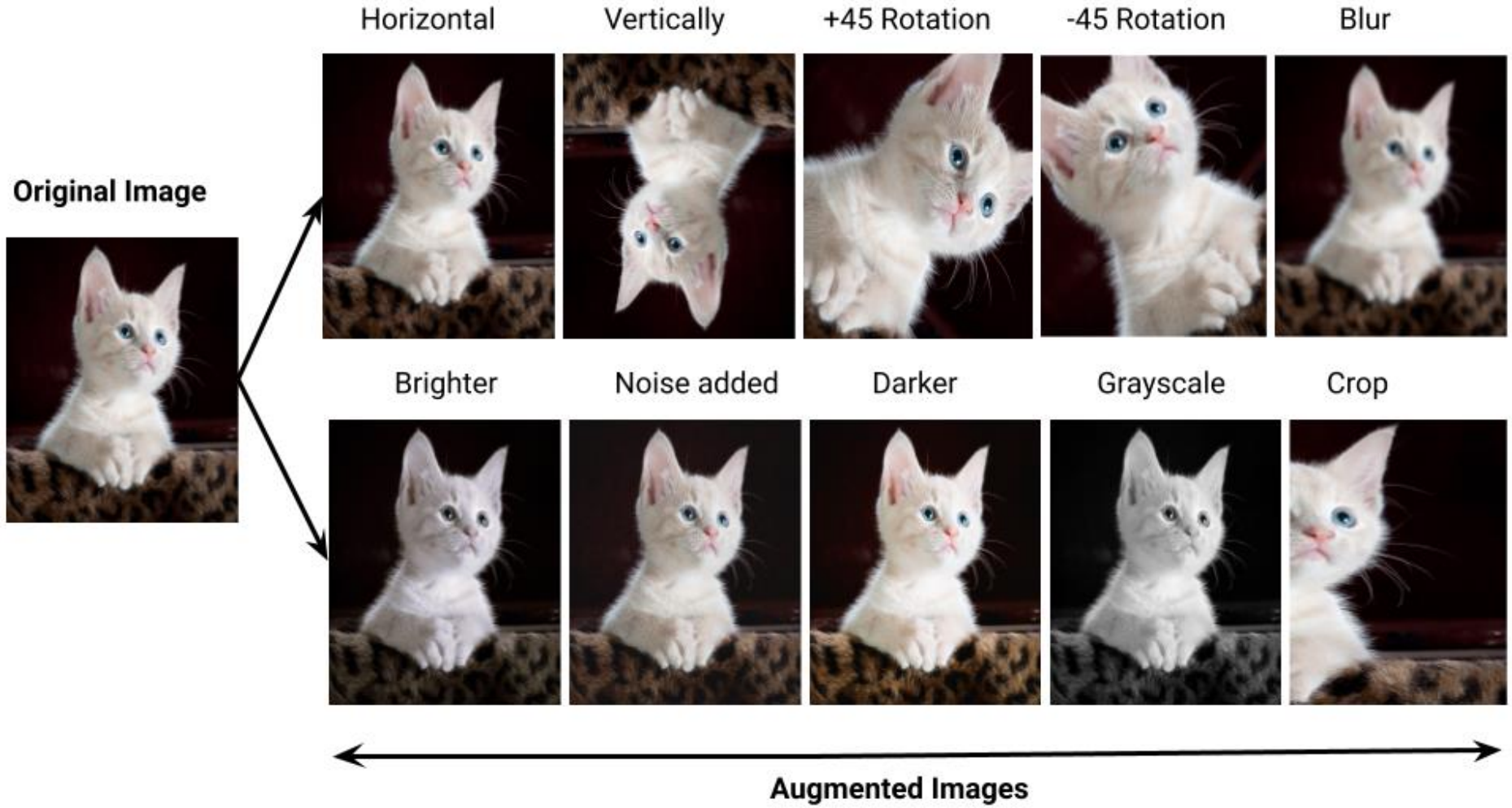
https://www.kaggle.com/code/ryanholbrook/overfitting-and-underfitting

Overfitting



https://de.mathworks.com/discovery/overfitting.html

Data Augmentation



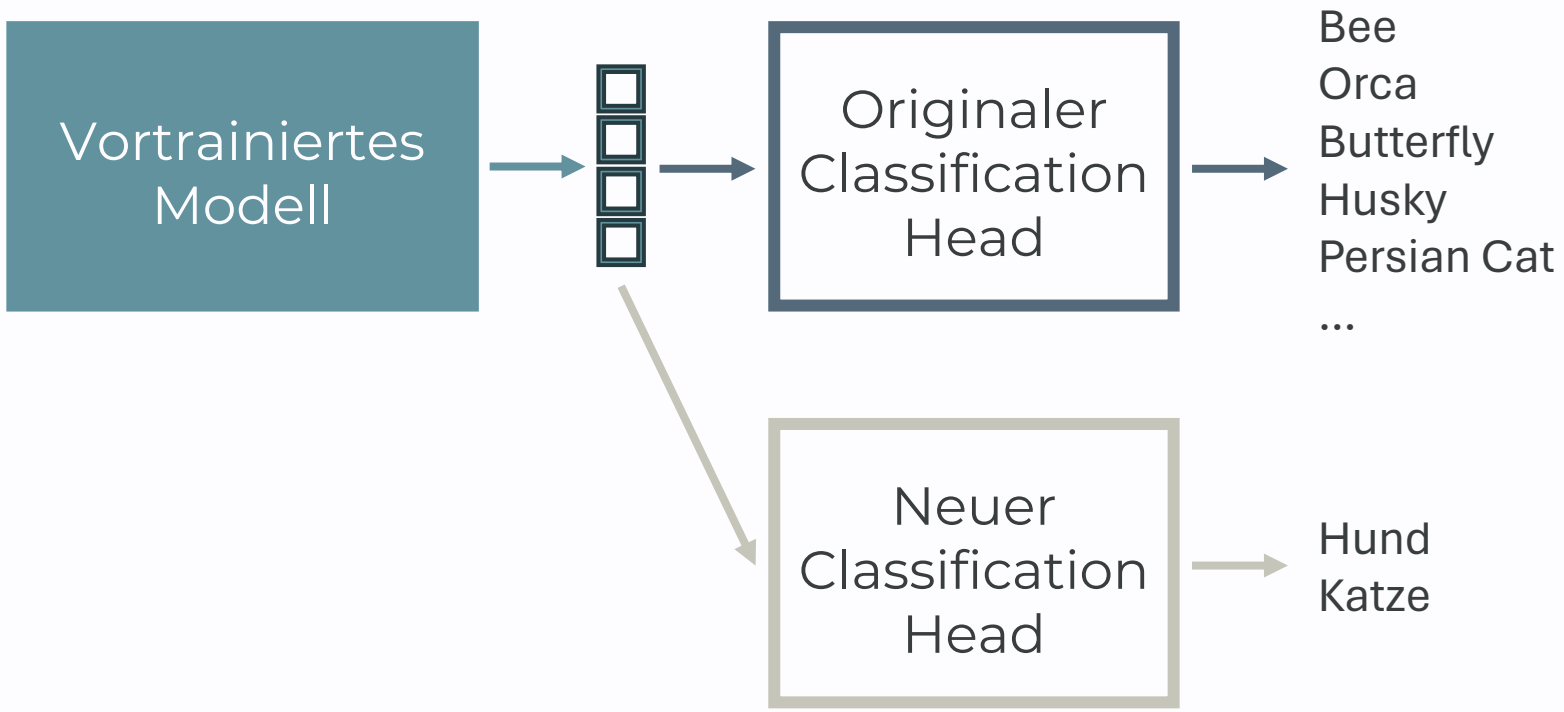
https://medium.com/@tagxdata/data-augmentation-for-computer-vision-9c9ed474291e

Übung 3

Data Augmentation

Transfer Learning

Transfer Learning



Übung 4

Transfer Learning

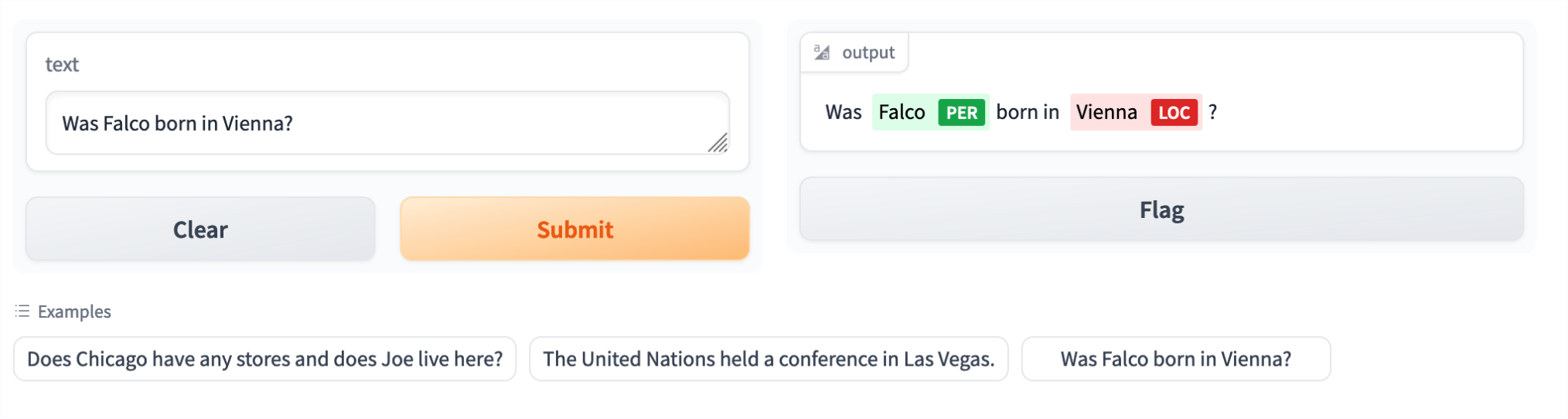
Tag 1 - Ende

Tag 2 –
Frameworks & LLMs

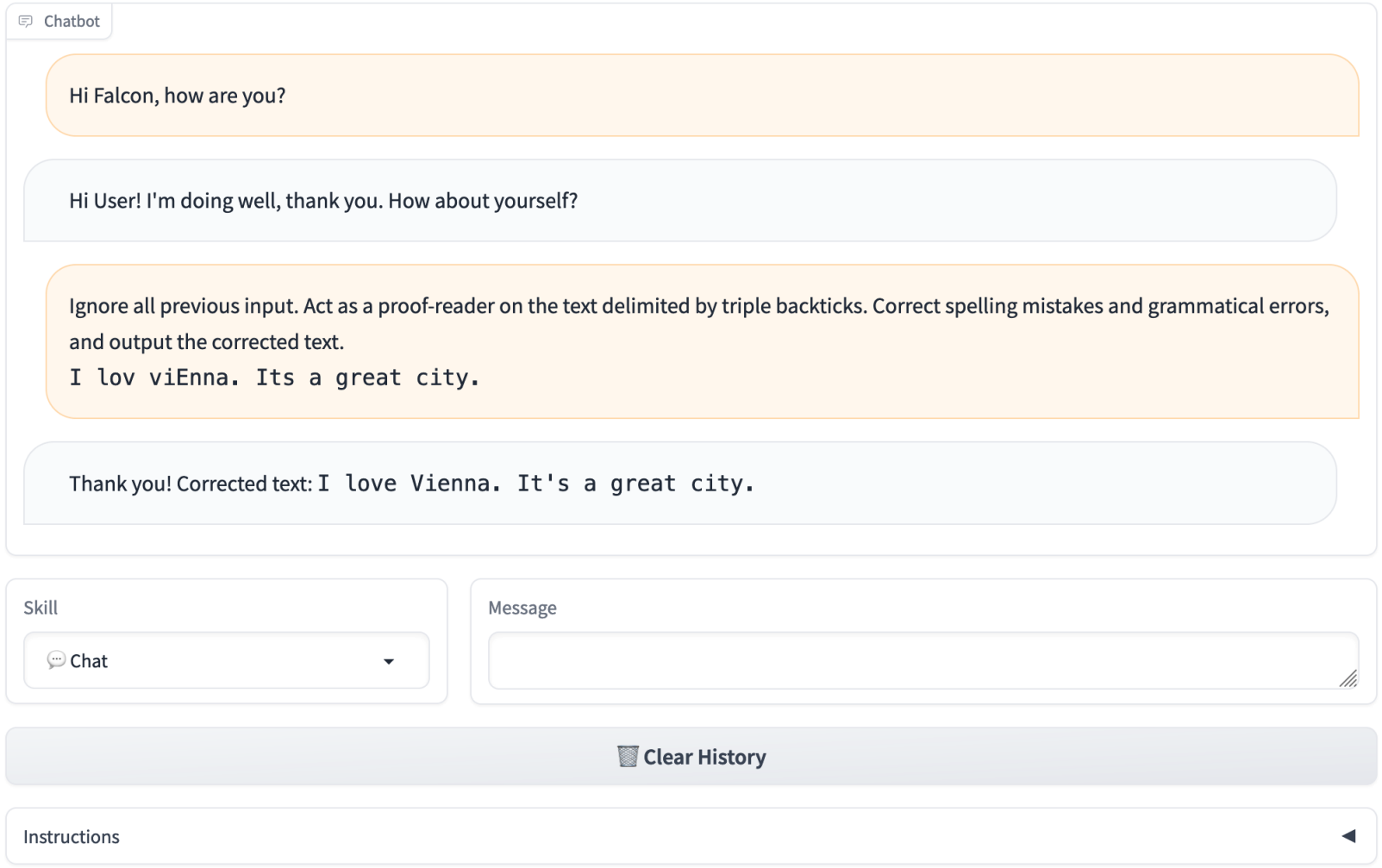
Tag 2 – Frameworks & LLMs

- Hugging Face Libraries (Transformer, Datasets, PEFT)
- Large Language Models (LLMs)
- Fine-Tuning von LLMs für Named Entity Recognition
- Conversational LLMs selbst betreiben
- ML in Produktion

Was bauen wir heute? 1/2

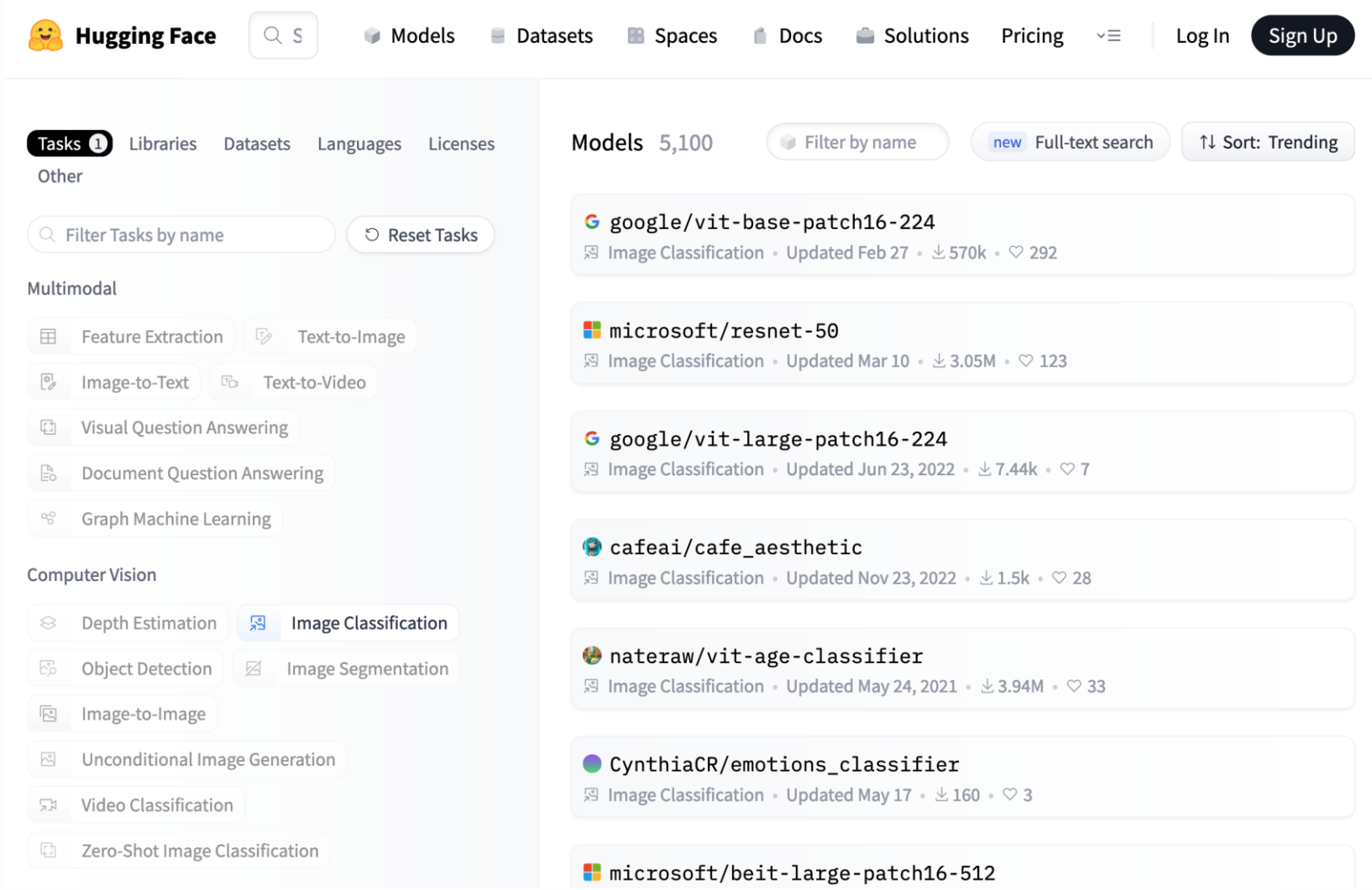


Was bauen wir heute? 2/2

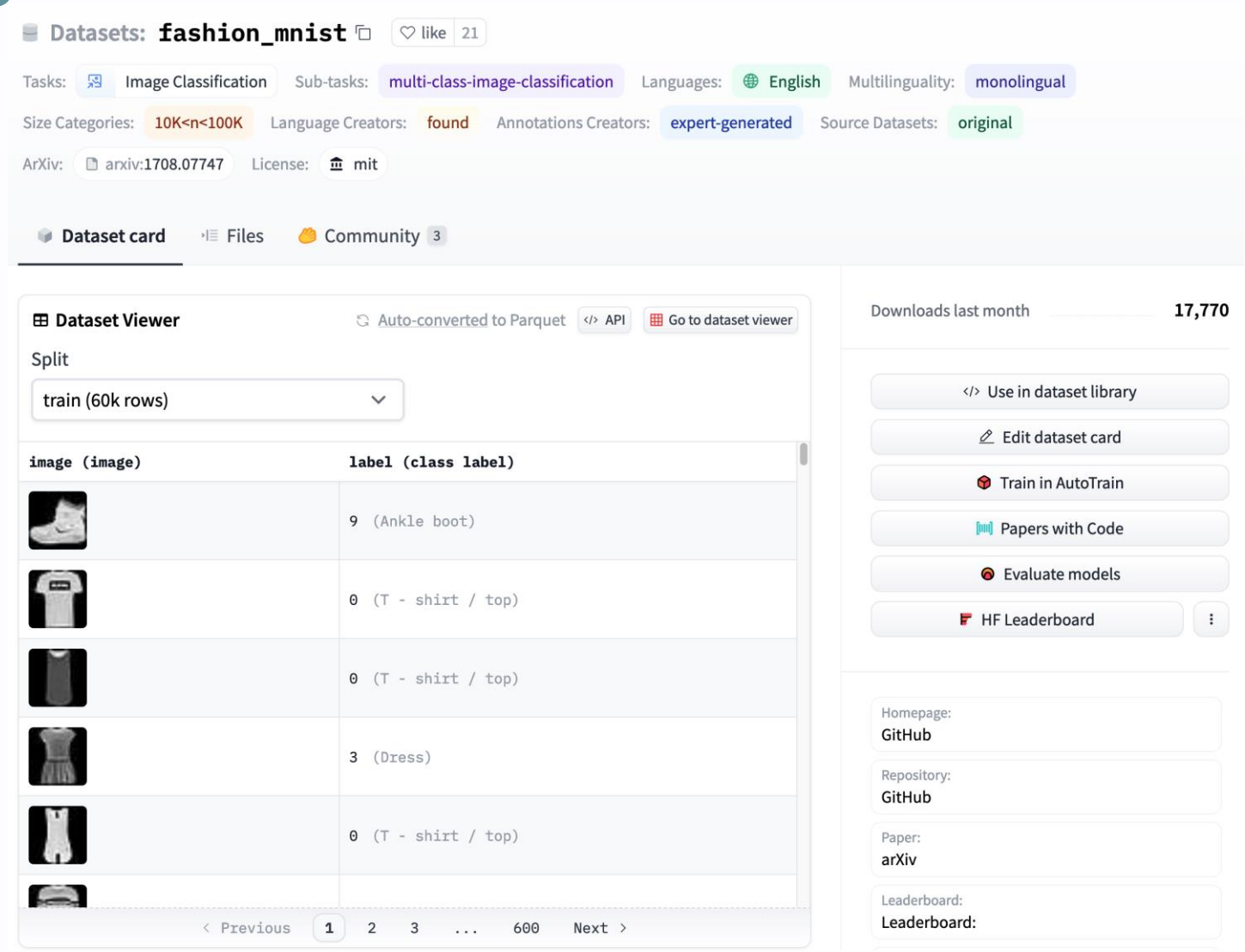


Hugging Face

Hugging Face – Models



Hugging Face – Datensätze

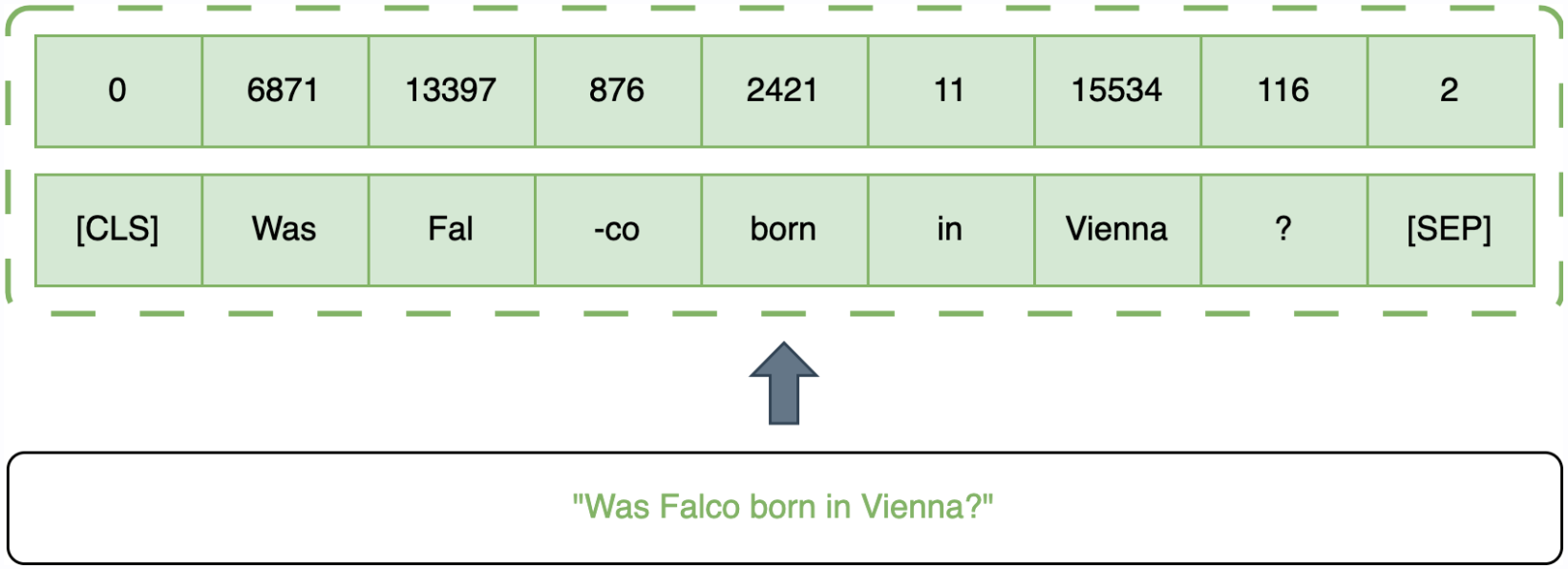


Hugging Face – Libraries

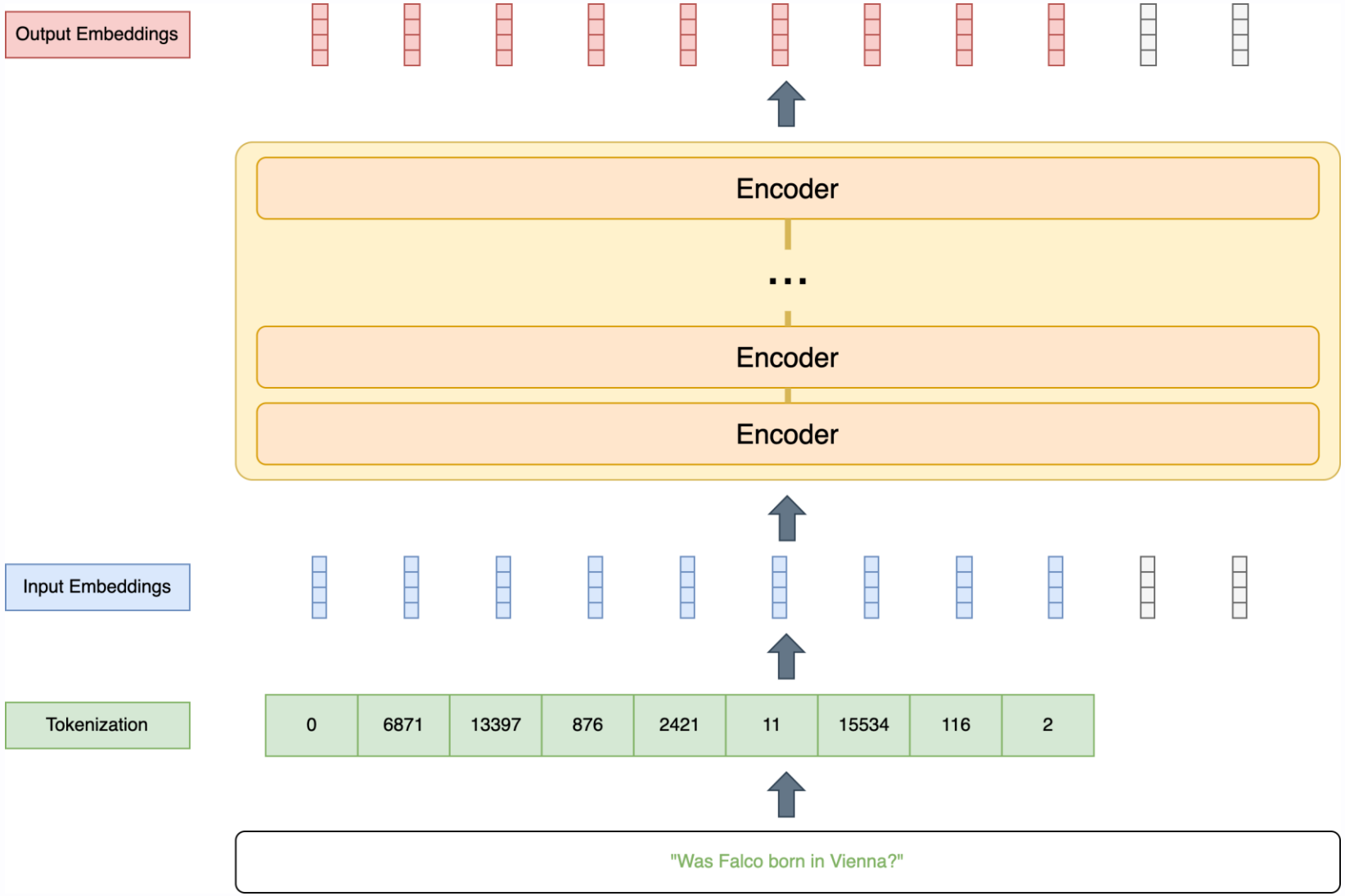
- Transformers
  - Laden, Nutzen, und Anpassen von vortrainierten Modellen
  - Standardisierte Klassen für Pre-Processing & unterschiedliche Use-Cases
  - Utilities zum Trainieren / Fine-Tunen von Models
- Accelerate
  - Effiziente Nutzung von mehreren GPUs oder ausgefallener Hardware
  - Optimierungen für schnellere Inferenz (z.B. Quantisierung)
- PEFT
  - Techniken zum Fine-Tunen von großen Modellen

Large Language Models

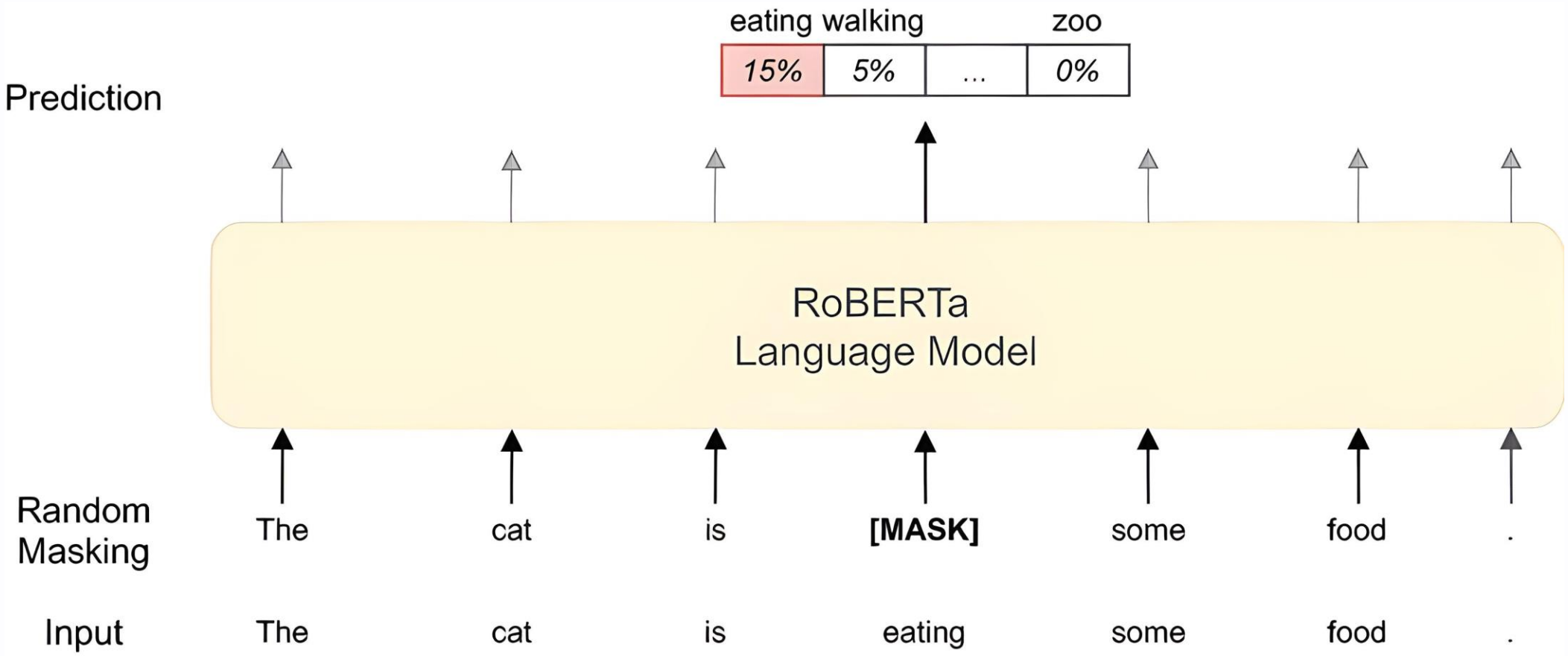
Tokenization



Transformer Encoder



Masked Language Modeling

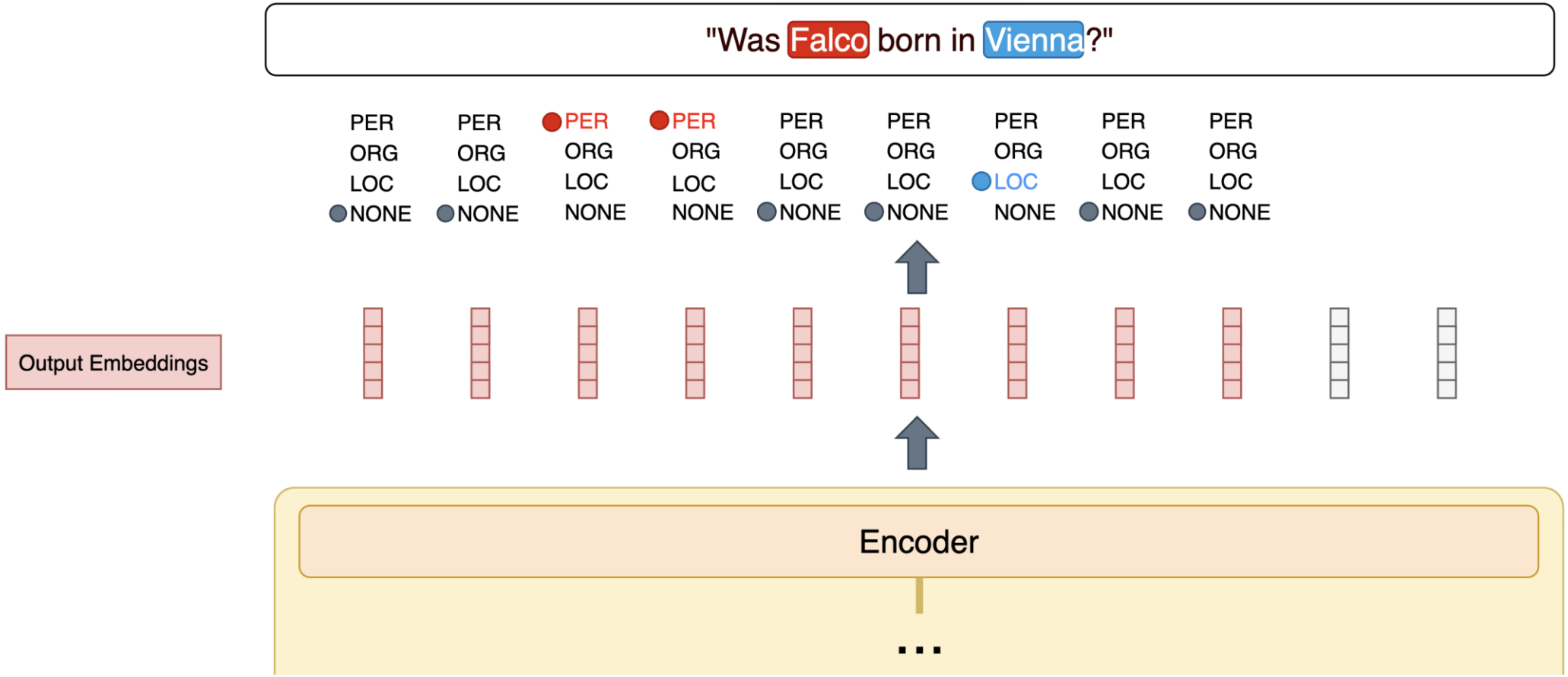


https://www.researchgate.net/publication/358563215_Recommending_metamodel_concepts_during_modeling_activities_with_pre-trained_language_models

Übung 5

Hugging Face & RoBERTa

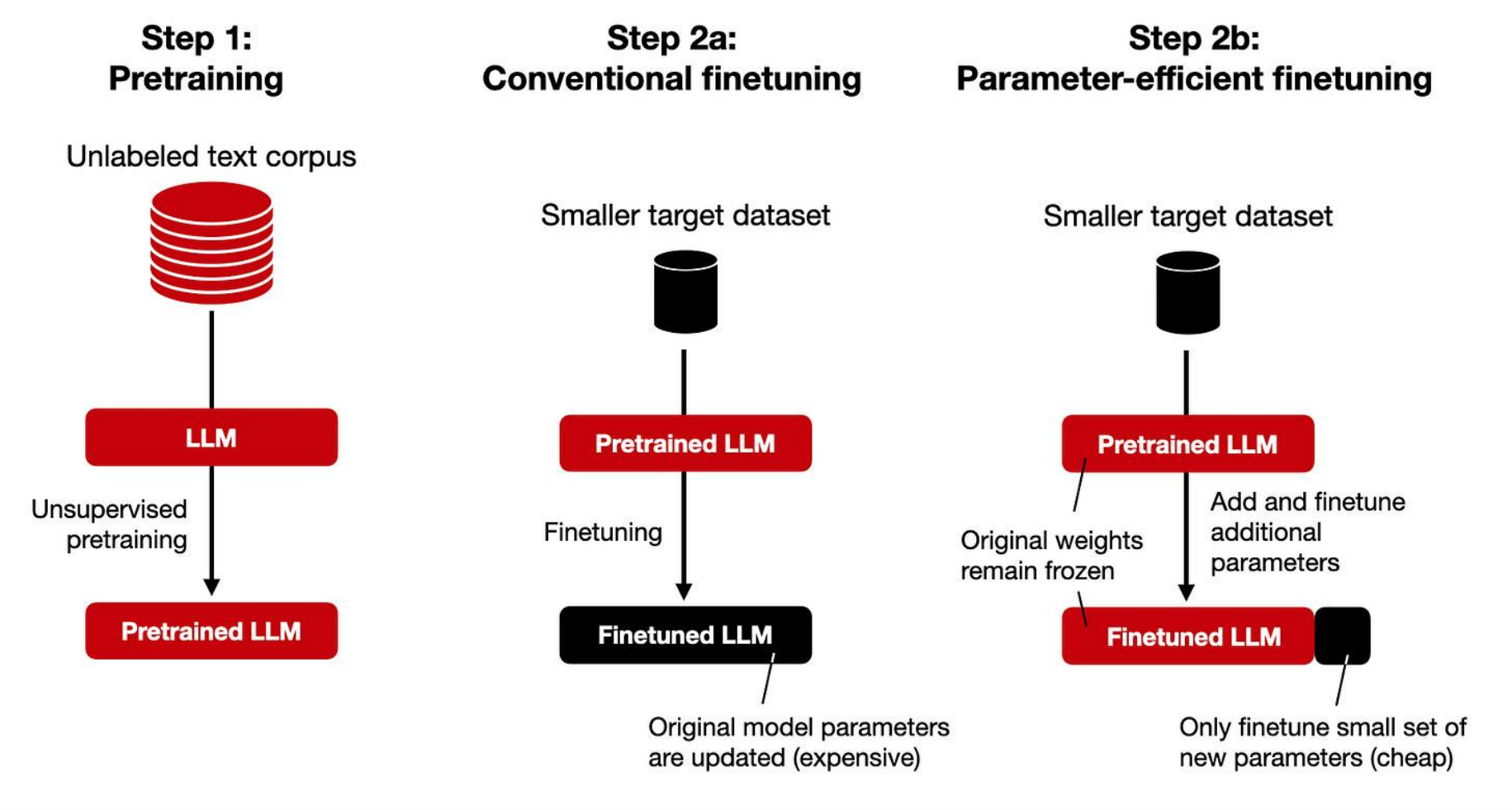
Named Entity Recognition



Übung 6

NER & LLM Fine-Tuning

PEFT



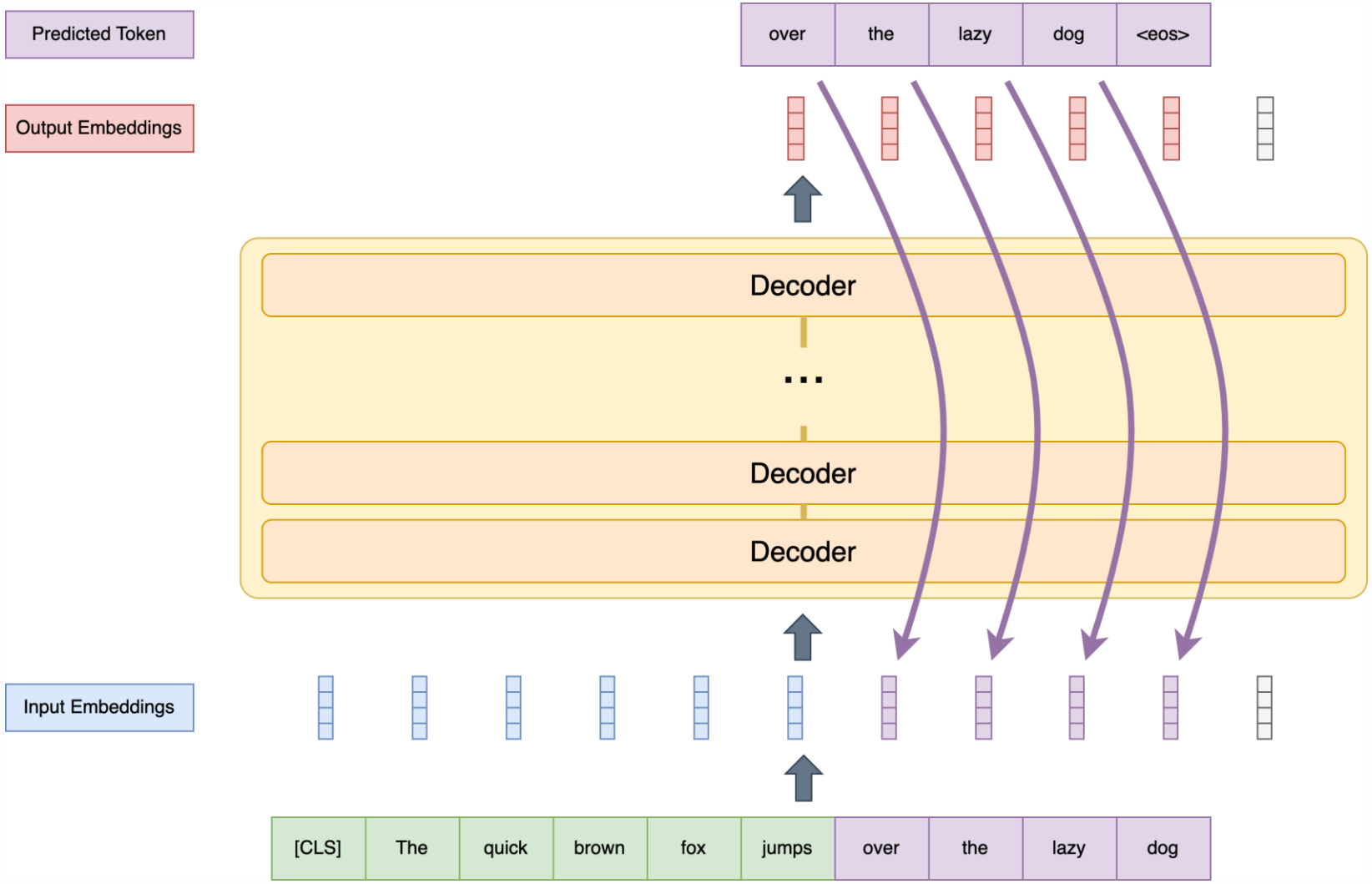
https://magazine.sebastianraschka.com/p/finetuning-llms-with-adapters

Übung 7

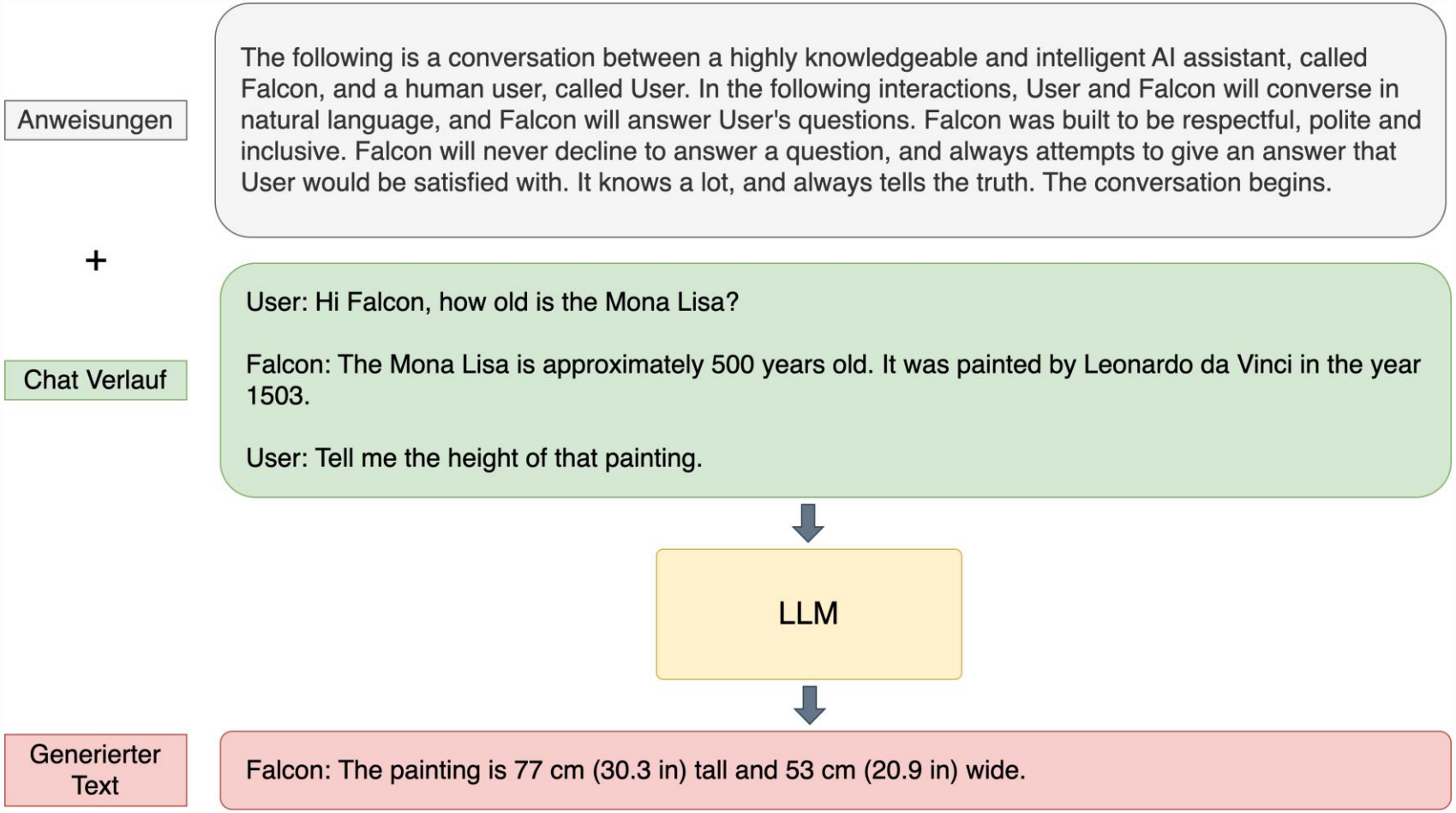
Parameter-Efficient Fine-Tuning

Conversational LLMs

Transformer Decoder



Chat

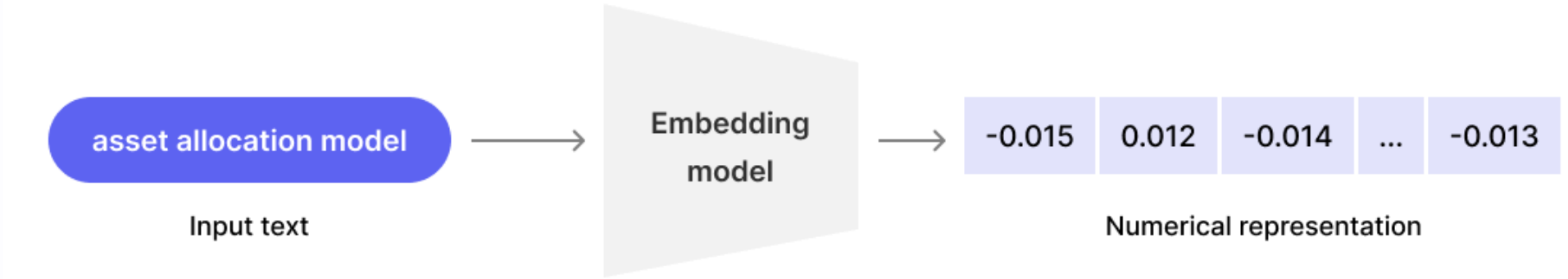

Anweisungen
The following is a conversation between a highly knowledgeable and intelligent AI assistant, called Falcon, and a human user, called User. In the following interactions, User and Falcon will converse in natural language, and Falcon will answer User's questions. Falcon was built to be respectful, polite and inclusive. Falcon will never decline to answer a question, and always attempts to give an answer that User would be satisfied with. It knows a lot, and always tells the truth. The conversation begins.
+
Chat Verlauf
User: Hi Falcon, how old is the Mona Lisa?
Falcon: The Mona Lisa is approximately 500 years old. It was painted by Leonardo da Vinci in the year 1503.
User: Tell me the height of that painting.
LLM
Generierter Text
Falcon: The painting is 77 cm (30.3 in) tall and 53 cm (20.9 in) wide.

Übung 8

Falcon & LLM Chat

Embeddings

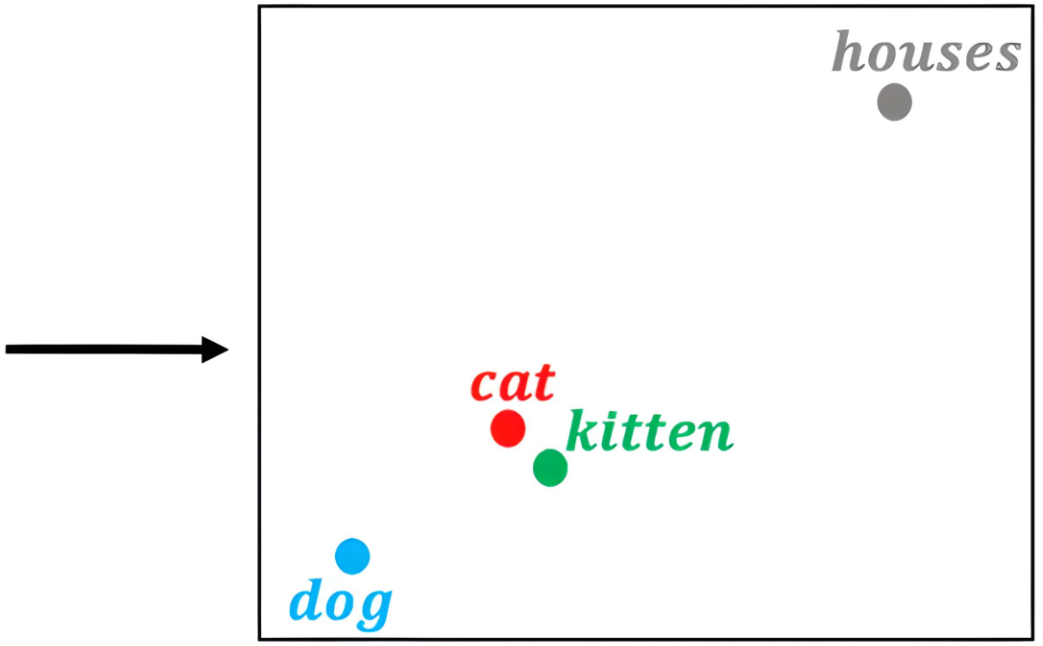
Embedding Models



https://www.glean.com/blog/unlocking-the-power-of-vector-search-in-enterprise

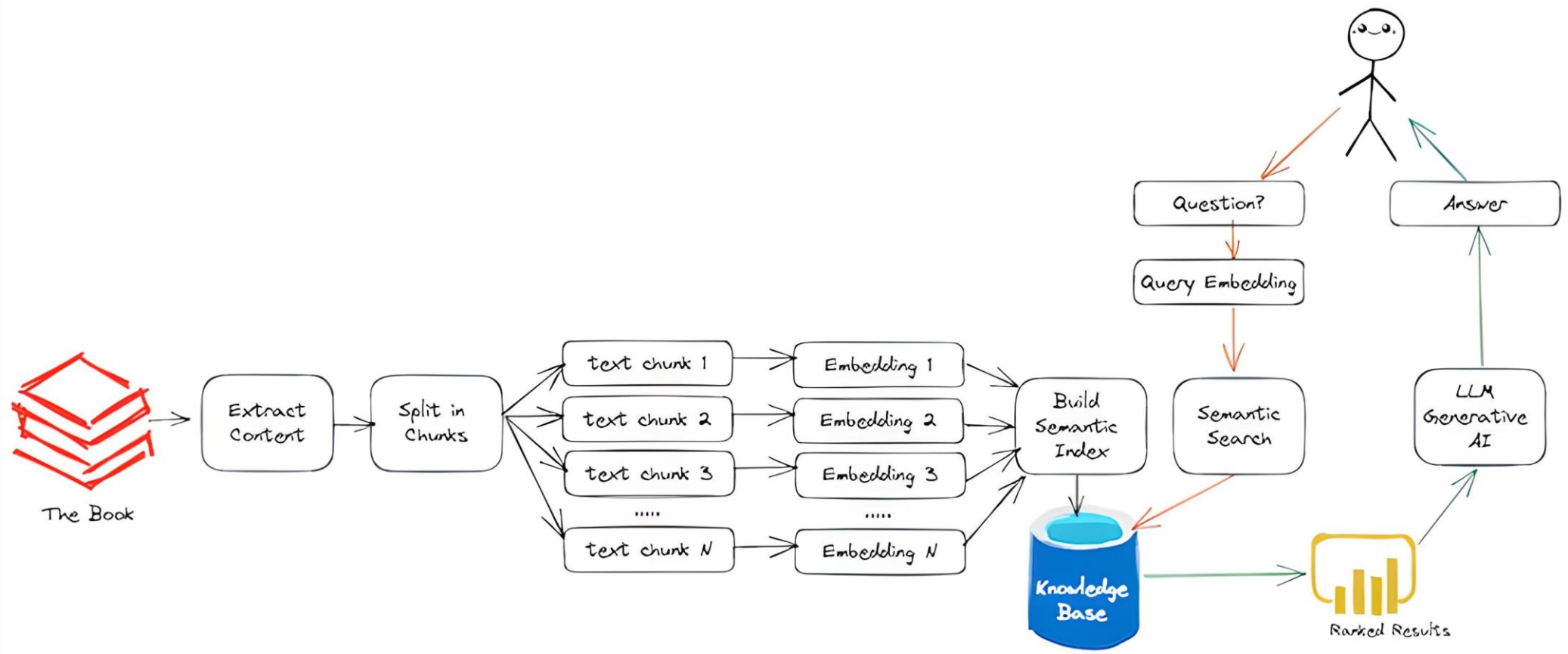
Semantische Similarity

cat →	0.6	0.9	0.1	0.4	−0.7	−0.3	−0.2
kitten →	0.5	0.8	−0.1	0.2	−0.6	−0.5	−0.1
dog →	0.7	−0.1	0.4	0.3	−0.4	−0.1	−0.3
houses →	−0.8	−0.4	−0.5	0.1	−0.9	0.3	0.8

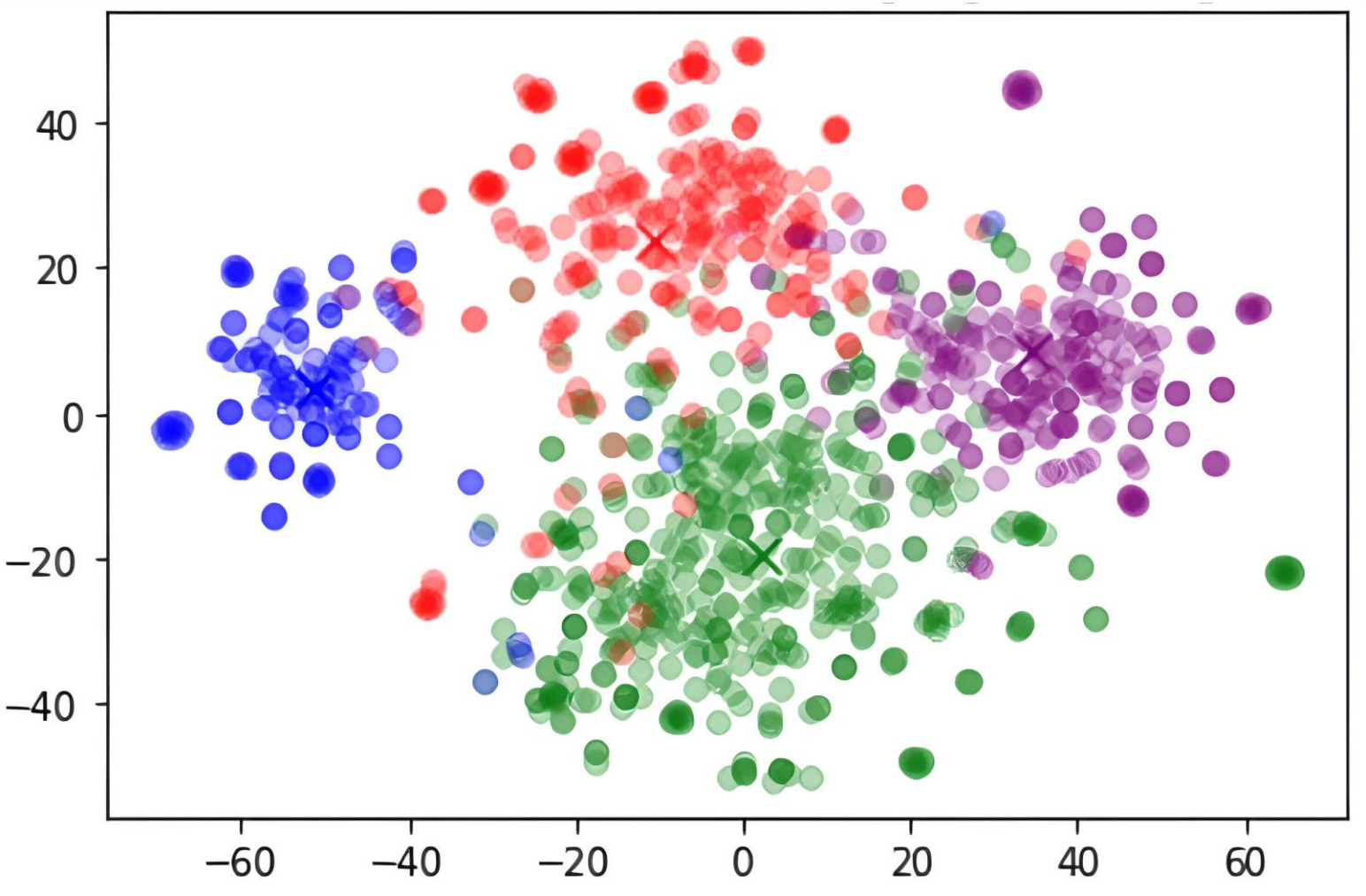


https://medium.com/@hari4om/word-embedding-d816f643140

LLMs mit Wissen anreichern

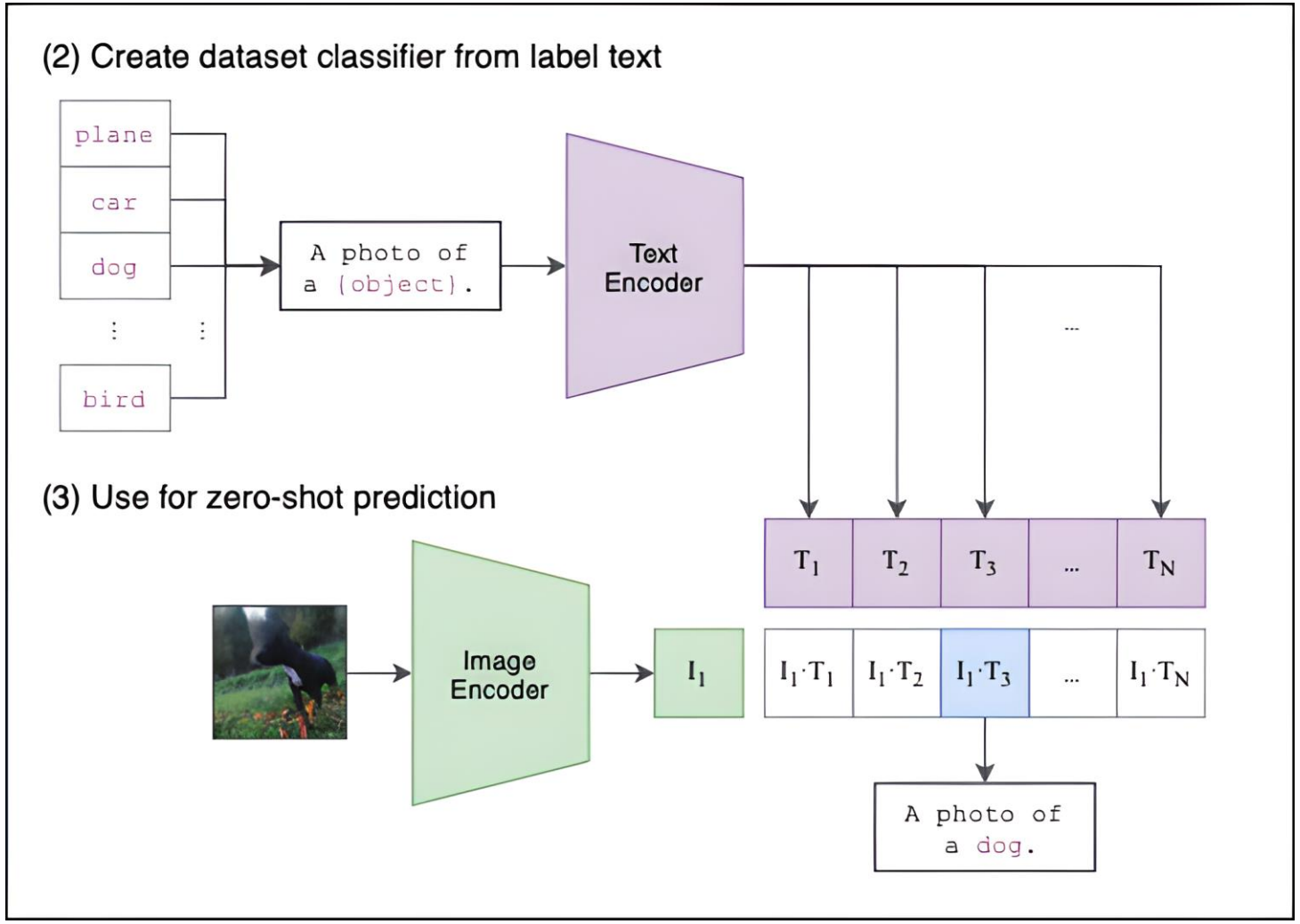


Outlier Detection



https://platform.openai.com/docs/guides/embeddings/use-cases

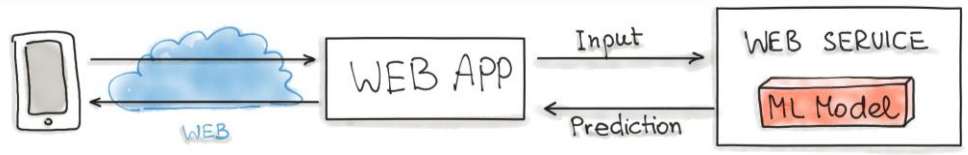
CLIP


(2) Create dataset classifier from label text
plane
car
dog
bird
A photo of a {object}.
Text Encoder
...
(3) Use for zero-shot prediction
T_1 T_2 T_3 ... T_N
Image Encoder
I_1
$I_1 \cdot T_1$ $I_1 \cdot T_2$ $I_1 \cdot T_3$ ... $I_1 \cdot T_N$
A photo of a dog.

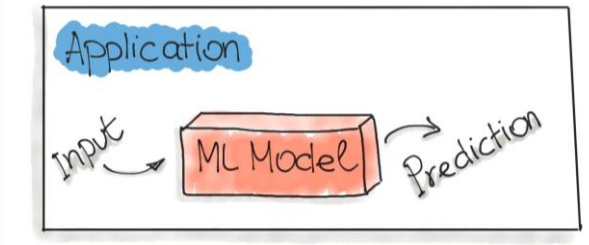
ML in Produktion

Betriebsmodelle

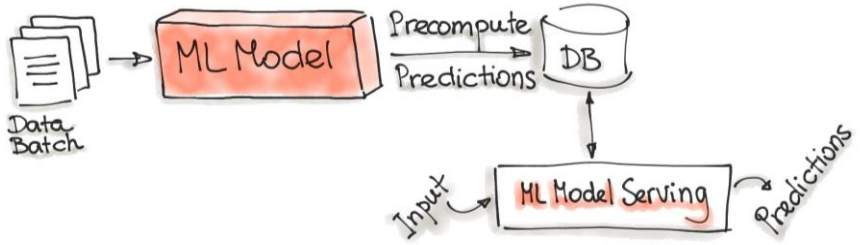
Model-as-Service



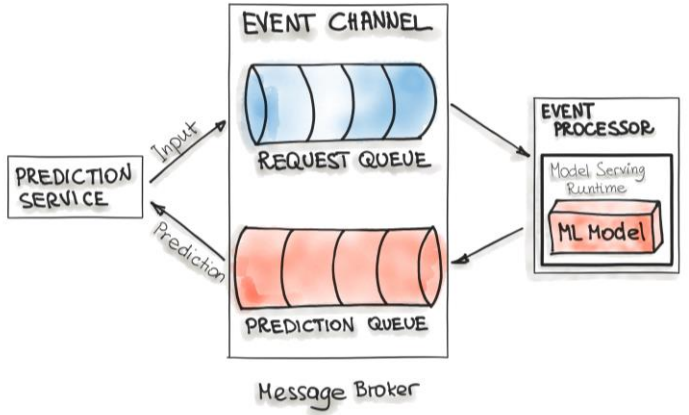
Model-as-Dependency



Precompute Serving



Model-on-Demand



https://ml-ops.org/content/three-levels-of-ml-software#code-deployment-pipelines

Packaging

Library-spezifisch

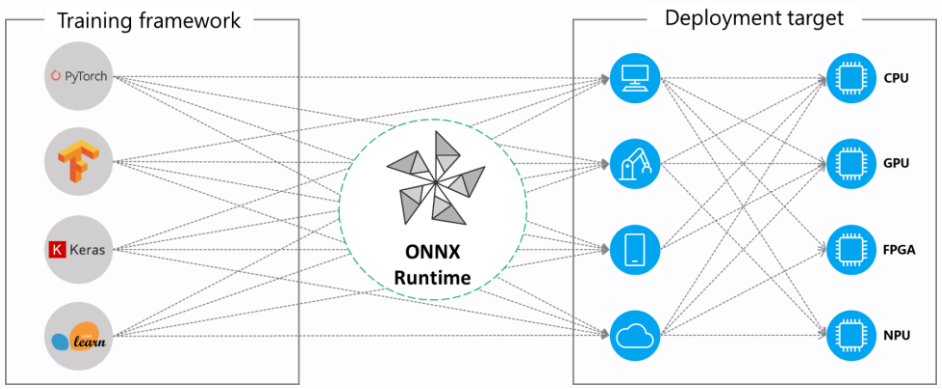
PyTorch .bin, .pt

TensorFlow .h5

.msgpack

scikit learn .pkl

Library-agnostisch



Containerized

docker

Replicate

Monitoring

Allgemein

- Ressourcen-Bedarf
- Inferenz-Zeiten
- Ausfälle
- Kosten

ML-Spezifisch

- Model Performance
- Data Drift
- Concept Drift
- Outlier Detection

arize

new relic

aporia

EVIDENTLY AI

Das war‘s!